POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Developing New High-Order Sequential Simulation Methods Based on Learning-Oriented Kernels

LINGQING YAO
Département de mathématiques et de génie industriel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*
Génie industriel

Août 2020

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée :

Developing New High-Order Sequential Simulation Methods Based on Learning-Oriented Kernels

présentée par **Lingqing YAO**
en vue de l'obtention du diplôme de *Philosophiæ Doctor*
a été dûment acceptée par le jury d'examen constitué de :

Luc ADJENGUE, président
Michel GAMACHE, membre et directeur de recherche
Roussos DIMITRAKOPOULOS, membre et codirecteur de recherche
Mustafa KUMRAL, membre
Julian ORTIZ CABRERA, membre externe

DEDICATION

In memory of my father Huailiang Yao (1958-2011).

ACKNOWLEDGEMENTS

I sincerely thank my supervisor, Prof. Michel Gamache, for his support in going through all the essential steps towards my Ph.D. program at Polytechnique Montreal, as well as allowing me to collaborate with COSMO Stochastic Mine Planning Lab, Dept. of Mining and Materials Engineering, McGill University in my research. I want to express my deepest gratitude to the other supervisor, Prof. Roussos Dimitrakopoulos, for his insightful guidance, patience, and encouragement throughout my research. His expertise in geostatistics and his promoting the pursuit of excellence always kept me motivated and performing my best. I feel fortunate to have been working in the COSMO Lab, directed by Prof. Dimitrakopoulos, who invariably strives to develop cutting-edge ideas and never hinders innovation.

I want to thank all my colleagues whom I worked with from COSMO Lab. A special thanks to my office-mate Ashish Kumar, for his friendship and the good suggestions that came out of our discussions. My thanks to Dr. Ilnur Minniakhmetov for his helpful insights and his development of programs to generate the cumulant maps. Thanks to Adrien, Amina, Amir, Christian, Cosmin, Daniel, Fernanda, Mélanie, Luis, Ryan, Ray, Yanyan, Zachary, and Zayneb all of whom filled my life in Montreal with many joyful moments, as well as all the former and new students at COSMO Lab. Life would have been much more difficult without Deborah's help in all the administrative matters at COSMO Lab. I owe my thanks to her and also to Diane, Melisa, and Saroja for their administration at Polytechnique. I want to thank all the sponsors who supported my Ph.D. research. This work is funded by the NSERC CRD Grant CRDPJ 500414-16, the COSMO mining industry consortium (AngloGold Ashanti, Barrick Gold, BHP, De Beers, IAMGOLD, Kinross, Newmont Mining and Vale), and NSERC Discovery Grant 239019.

I want to thank my Mum, Xiaomei Xu, and my brothers, Lingyun and Lingxiao, who unconditionally supported me as they always have. I especially thank my wife Xiang, my two sons, Fengming and Yiming, who have accompanied and loved me throughout.

I am grateful to Dr. Ted Kass and Margaret for their teaching me the eternal truth from the Bible and leading me to faith in Jesus Christ. Their genuine concern and sincere prayers warmed my heart. My thanks also go to pastor Hao Wu from the Montreal Chinese Alliance Grace Church for his preaching, caring, and prayers, and all the brothers and sisters from the fellowship for their love and prayers. "And we know that all things work together for good to those who love God, to those who are the called according to His purpose." (Romans 8:28)

RÉSUMÉ

La quantification de l'incertitude joue un rôle essentiel dans la gestion du risque technique de l'exploitation durable des ressources naturelles. Les modèles de champs aléatoires sont utilisés pour modéliser les attributs naturels d'intérêt, parmi lesquels les attributs à différents endroits sont représentés comme des variables aléatoires comprenant une distribution de probabilité conjointe. Les statistiques spatiales, qui varient selon différents modèles de champs aléatoires, décrivent mathématiquement les structures spatiales. Les méthodes de simulation stochastique génèrent des réalisations multiples basées sur certains modèles de champs aléatoires afin de représenter les résultats possibles des attributs naturels considérés. Elles visent à reproduire les statistiques spatiales des données perçues, fournissant ainsi des outils utiles pour quantifier l'incertitude spatiale des attributs cibles. Dans le contexte des applications minières, la reproduction de structures spatiales, à partir des données-échantillons, a un impact significatif sur la gestion des risques liés aux décisions de planification minière. Plus précisément, la valeur actualisée nette (VAN) d'un gisement minéral, compte tenu d'un calendrier de planification minière donné, dépend des revenus générés par les séquences d'extraction des matériaux souterrains, les flux de trésorerie étant actualisés en fonction des périodes d'exploitation. Les séquences d'extraction des matériaux, à leur tour, sont déterminées par la distribution spatiale des teneurs en métaux, en particulier la continuité spatiale des éléments métalliques enrichis.

Les méthodes de simulation stochastique d'ordre élevé ne présupposent aucune distribution de probabilité spécifique sur les modèles de champs aléatoires, évitant ainsi les limites des modèles de champs aléatoires gaussiens traditionnels. De plus, ces méthodes tiennent compte des statistiques spatiales d'ordre élevé qui caractérisent les interactions statistiques entre les attributs aléatoires en de multiples endroits et elles ont donc l'avantage de reproduire des structures spatiales complexes. Par conséquent, cette thèse développe de nouvelles méthodes de simulation stochastique d'ordre élevé basées sur un cadre proposé d'apprentissage statistique et de noyaux orientés sur l'apprentissage, visant à faire progresser les aspects théoriques des méthodes de simulation stochastique ainsi que les aspects pratiques des décisions minières sous incertitude. Le paradigme général de la simulation séquentielle est adopté dans cette thèse afin de générer des réalisations à partir de modèles de champs aléatoires, ce qui décompose les distributions de probabilités conjointes en une séquence de distributions de probabilités conditionnelles.

La méthode originale de simulation d'ordre élevé utilise la série d'expansions du polynôme

de Legendre pour l'approximation des distributions de probabilités conjointes des champs aléatoires. Les coefficients de la série d'expansion du polynôme sont dérivés du calcul de ce que l'on appelle les cumulants spatiaux qui doivent être stockés dans une structure arborescente en mémoire et certains termes de la série du polynôme sont abandonnés compte tenu de la complexité du calcul. À titre de première contribution, un nouveau modèle de calcul de simulation d'ordre élevé est ici proposé, évitant le calcul explicite des cumulants spatiaux et le stockage des résultats calculés. Une fonction unifiée est dérivée comme une forme d'équivalence à la série d'expansions du polynôme de Legendre sans abandonner aucun terme, tout en simplifiant les calculs en temps polynomial. La méthode de simulation proposée conduit à un algorithme récursif de dérivation de la distribution de probabilités conditionnelles. À titre de deuxième contribution, une nouvelle fonction du noyau, ce qu'on appelle le noyau spatial du moment de Legendre, est proposée pour intégrer des statistiques spatiales d'ordre élevé des données originales dans le nouvel espace du noyau. Un cadre d'apprentissage statistique est proposé pour découvrir la distribution de probabilité cible du champ aléatoire en la faisant correspondre aux statistiques spatiales d'ordre élevé observées dans les données disponibles grâce à un algorithme à noyau. Le nouveau cadre d'apprentissage statistique pour la simulation d'ordre élevé a la capacité de généralisation nécessaire pour atténuer les conflits statistiques entre les données-échantillons et l'image d'entraînement, comme le confirment les études de cas avec des données synthétiques. Un gisement d'or tridimensionnel est réalisé pour montrer ses aspects pratiques dans une mine réelle, en démontrant la reproduction de statistiques spatiales d'ordre élevé à partir des données-échantillons de forage.

Pour éviter l'impact d'éventuels conflits statistiques avec les données-échantillons en utilisant une image d'entraînement, une méthode de simulation d'ordre élevé sans image d'entraînement est développée en se basant sur le cadre d'apprentissage statistique ci-dessus. Une nouvelle approche d'agrégation de noyaux est proposée afin de permettre la découverte de données éparses. Les événements de données, comme les données de conditionnement, correspondent aux valeurs d'attribut associées aux modèles spatiaux de diverses configurations géométriques. L'agrégation de noyaux combine l'ensemble des éléments dans différents sous-espaces du noyau pour l'inférence statistique, en utilisant efficacement les informations incomplètes des répliques qui correspondent partiellement au modèle spatial d'un événement de données spécifique. L'étude de cas montre une bonne reproduction des statistiques spatiales d'ordre élevé des données-échantillons sans utiliser les images d'entraînement.

Notre dernière contribution vise à atteindre la distribution de probabilité cible des modèles de champs aléatoires en apprenant des informations spatiales d'ordre élevé provenant de différentes sources à différentes échelles. Plus précisément, l'agrégation de noyaux est proposée pour incorporer les statistiques spatiales d'ordre élevé à une échelle grossière à partir

des données-échantillons et pour compléter les statistiques spatiales d'ordre élevé à petite échelle à partir de l'image d'entraînement. De plus, un logiciel est développé et décrit pour faciliter les applications. Des études de cas, dans un gisement d'or et avec un ensemble de données synthétique, sont menées respectivement afin de tester la méthode et le programme développé.

ABSTRACT

Uncertainty quantification plays a vital role in managing the technical risk of the sustainable exploitation of natural resources. Random field models are utilized to model the natural attributes of interest, within which the attributes at different locations are represented as random variables comprising a joint probability distribution. Spatial statistics, varied with different random field models, mathematically describe the spatial patterns. Stochastic simulation methods generate multiple realizations based on certain random field models to represent the possible outcomes of natural attributes under consideration. They aim to reproduce spatial statistics of the perceived data, thus providing useful tools to quantify the spatial uncertainty of the target attributes. In the context of mining applications, reproducing spatial patterns from the sample data has a significant impact on managing the risks of mine planning decisions. Specifically, the net present value (NPV) regarding a certain mine planning schedule of a mineral deposit depends on the revenue generated by the extraction sequences of the underground materials, as the cash flows are discounted by the mining periods. The extraction sequences of the materials, in turn, are driven by the spatial distributions of metal grades, especially the spatial continuity of enriched metal elements.

High-order stochastic simulation methods make no assumption of any specific probability distribution on the random field models, avoiding the limitation of traditional Gaussian random field models. In addition, the methods account for the high-order spatial statistics that characterize the statistical interactions among random attributes at multiple locations and thus have the advantage of reproducing complex spatial patterns. Therefore, this thesis develops new high-order stochastic simulation methods based on a proposed framework of statistical learning and learning-oriented kernels, aiming to advance the theoretical aspects of the stochastic simulation methods, as well as the practical aspects of mining decisions under uncertainty. The general paradigm of sequential simulation is adopted in this thesis to generate realizations from the random field models, which decomposes the joint probability distributions into a sequence of conditional probability distributions.

The original high-order simulation method uses the Legendre polynomial expansion series for the approximation of the joint probability distributions of the random fields. The coefficients of the polynomial expansion series are derived from the computation of so-called spatial cumulants, which have to be stored in a tree structure in memory. In addition, some terms from the polynomial series are dropped considering the computational complexity. As a first contribution, a new computational model of high-order simulation is proposed herein,

which avoids the explicit computation of spatial cumulants and the storage of the computed results. A unified function is derived as the equivalency form to the Legendre polynomial expansion series without dropping out any terms, while simplifying the computations to polynomial time. The proposed simulation method leads to a recursive algorithm of deriving the conditional probability distribution.

As a second contribution, a new kernel function, the so-termed spatial Legendre moment kernel, is proposed to embed high-order spatial statistics of the original data into the new kernel space. A statistical learning framework is proposed to learn the target probability distribution of the random field by matching the expected high-order spatial statistics with regard to the target distribution to the observed high-order spatial statistics of the available data through a kernelized algorithm. The new statistical learning framework for high-order simulation has the generalization capacity to mitigate the statistical conflicts between the sample data and the training image, as confirmed by the case studies with a synthetic data set. Case study at a three-dimensional gold deposit shows the practical aspects of the proposed method in a real-life mine, demonstrating the reproduction of high-order spatial statistics from the drill-hole sample data.

To avoid the impact of potential statistical conflicts with the sample data by using a training image, a training-image free high-order simulation method is developed based on the above statistical learning framework. A new concept of aggregated kernel statistics is proposed to enable sparse data learning. The data events, as the conditioning data, correspond to the attribute values associated with the so-called spatial templates of various geometric configurations. The aggregated kernel statistics combine the ensemble of the elements in different kernel subspaces for statistical inference, efficiently utilizing the incomplete information from the replicates, which partially match to the spatial template of a given data event. The case study shows an effective reproduction of the high-order spatial statistics of the sample data without using the TI.

Our last contribution aims to achieve the target probability distributions of the random field models by learning high-order spatial information from different sources at multiple scales. Specifically, the aggregated kernel statistics is proposed to incorporate the high-order spatial statistics at coarse-scale from the sample data and to complement the high-order spatial statistics at fine-scale from the TI. In addition, a software is developed and described to facilitate the applications. Case studies with a synthetic data set and at a gold deposit are conducted respectively to test the method and the developed program.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

CCDF	Conditional Cumulative Distribution Function
CDF	Cumulative Distribution Function
CPDF	Conditional Probability Density Function
HOSIM	High-order Stochastic Simulation
MPS	Multiple Point Simulation
RKHS	Reproducing Kernel Hilbert Space
SLM	Spatial Legendre Moment
TI	Training Image

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

1.1 Overview

Uncertainty quantification is important in modeling natural phenomena in complex Earth systems. The perceived attributes from various natural phenomena exhibit randomness in their spatial structures or patterns as the related natural systems evolved over considerable time spans and often date to the ancient era of the Planet Earth. The random field models are used to describe the randomness of the spatial attributes of natural phenomena regarding the complexity of the behind system dynamics. A random field $\mathbf{Z}(\mathbf{u})$ describes a stochastic model of the attributes of interest $\mathbf{Z}$ as random functions of locations $\mathbf{u}$. In other words, for a given set of locations $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\} \in \mathbb{R}^d$, $d = 1, 2$, or 3, depending on the spatial dimension, the random variables at these corresponding locations from the random field $\mathbf{Z}(\mathbf{u})$, $Z(u_i)(i = 1, \ldots, n)$, comprise a joint probability distribution. To quantify the uncertainty of a specific attribute at a certain location, one needs not only to characterize the proportion from its local probability distribution but also its statistical interactions in space with the related attributes at other locations. The statistical interactions among random variables at different locations are usually captured by the so-termed spatial statistics, and in general, appear as the spatial patterns in the related natural attributes. Stochastic simulation provides a tool for building the relevant random field models from the perceived earth science data. As a result, the spatial uncertainty is quantified by the so-called realizations generated from the random field models to represent the possible spatial distributions of the natural attributes.

The uncertainty quantification itself is not the ultimate goal in practice, but acts as an important input for risk assessment and decision making in engineering applications. The stochastic orebody modeling and mine planning decisions are specifically discussed herein to highlight the importance of uncertainty quantification in the context of mining applications. Nevertheless, it should be noted that the stochastic simulation methods developed in this thesis are general and applicable to other engineering fields. Mine planning comprises a collection of decisions at different mining stages through the available information to optimize the profit on investment. Generally, mine planning optimization can be categorized as constrained optimization problems in the scope of mathematical programming, where the most favourable solutions achieve the highest net present value (NPV), and the constraints are various factors related to mining activities, such as operational feasibility, quality constraints imposed by the processing plants, and other management policies, and so on. The very first step of the above optimization process starts from orebody modelling, through which com-

puterized representations of orebodies in three-dimensional space are provided as the input of the desired mine planning optimization. The orebody models are usually represented by a set of mining blocks discretised in the space of certain mining sites, and each block can be located by its coordinates in a grid and the related attributes, such as grades and material types, are attached. It is practically infeasible to take measurements of all the blocks in an orebody model to acquire the geological attributes of them. As a matter of fact, these attributes (associated with each block) are often inferred from the limited observations from the drilling samples, which is a reasonable approach considering the high costs to obtain the underground information.

Given the inputs from the orebody model and other parameters, such as costs incurred from mining activities and commodity prices from the market, the mining process can be regarded as a specialized *transfer function*. Meanwhile, the outputs of the transfer function, or namely *responses* to the inputs, correspond to the mining objectives, which may vary under different circumstances. There are also some undetermined decision variables that feed into the transfer function, which are called *parameters of interest*. These parameters decide the implementation of the optimal plan to achieve certain goals regarding the mining process. From this point of view, the main task of mine planning optimization is analogous to inverting the parameters of interest from intensifying responses of transfer functions in the context of mining processes, which is often solved by operational research methods. Ideally, the optimal selection of parameters of interest can be acquired by searching the entire solution spaces, given that the inputs are definite. However, in mining practice, the deterministic orebody models are not sufficient to represent the realistic ore reserves, and neither can the fluctuations of commodity price be reflected in a deterministic way. Considerable risk in decision making emerges because of uncertainty in the input parameters, since the parameters of interest are usually sensitive to the variation of inputs. The situation is like that of a random signal that triggers an uncertain response.

The uncertainty of orebody modelling is because only limited information of an ore reserve is revealed, usually from the sampling of drill holes. Thus, inference is needed to complete the modelling. For decades, the orebody models as inputs are generated by estimations with various interpolation methods in the traditional framework of mine planning optimization, with negligence of existing geological uncertainty. However, most of the interpolation methods are essentially a moving average and turn out to smooth the outputs by reducing the proportion of highest and lowest values. The deficiency of a deterministic orebody model by estimation is obvious when it is used in mine planning optimization, since, generally, the related mining process is a non-linear transfer function, and the average of inputs do not necessarily lead to the average of outcomes associated with the inputs. To reiterate mathematically in a rather

simplified way, suppose the estimation of parameters p are from some weighted average of possible values $p_1, \ldots, p_n$, i.e., $p = \sum_{i=1}^{n} w_i p_i$, where $w_1, \ldots, w_n$ are weights for each parameter, and let the transfer function be $f(\cdot)$, then the equation $\sum_{i=1}^{n} f(w_i p_i) = \sum_{i=1}^{n} w_i f(p_i)$ generally does not hold if $f(\cdot)$ is a non-linear function.

The risk of decision making in mining projects under geological uncertainty has been regarded the major factor behind not meeting project expectations and eventual project failure. For instance, Vallee [1] reports that 60% of mines surveyed have an average rate of production of less than 70% of the designed capacity in the first year of production. Instead of misrepresenting the orebody model through a single estimation, which is "precisely wrong", stochastic simulation provides an effective way to quantify geological uncertainty by representing the orebody models with a set of realizations of equal probability. In this way, risk assessment can be applied to reflect the fluctuation of financial forecasts of a certain mining project by implementing the plan under possible scenarios of an ore reserve situation. The risk analysis of traditional mine planning optimization from various publications further confirms the substantial deviation of their financial outcomes from the expectation. Ravenscroft [2] performs a risk analysis on a conventional mine planning schedule and shows that the chances of deviation from the expected metal grade within the range of 10% is less than 40% in probability. And as a practical mining project in Dowd [3] demonstrates, there is only 50% possibility of a conventional mining schedule to achieve a base-case NPV and the mean payback period is also greater than the expectation. Dimitrakopoulos et al. [4] take risk assessment on a traditional optimization study of a low-grade, epithermal, gold deposit, showing that there is a 95% probability of the project returning a lower NPV than predicted and, in the worst case, the NPV is 45% lower than the expected.

Clearly, geological uncertainty has a crucial impact on the financial outcome of a mining project, whereas conventional optimizers using a single deterministic orebody model are unable to deal with the adverse effect of the uncertainty. Dimitrakopoulos and Ramazan [5] propose a stochastic integer programming (SIP) framework of mine production scheduling optimization, which directly integrates the geological uncertainty into the objective function as the following:

$$\max \sum_{t=1}^{p} \Bigg[\underbrace{\sum_{i=1}^{n} E\Big\{(NPV)_i^t\Big\} b_i^t}_{\text{Part A}} - \underbrace{\sum_{s=1}^{m} (c_u^{to} d_{su}^{to} + c_l^{to} d_{sl}^{to} + c_u^{tg} + c_l^{tg} d_{sl}^{tg})}_{\text{Part B}} \Bigg] \tag{1.1}$$

where p is the total production period, n is the number of blocks, and b_i^t is a binary decision variable indicating whether a block i is mined during period t. The variables d and c represent the deviation from production targets and the unit cost for deviation, respectively, while the

subscripts u and l stand for the deviations or costs from excessive production (upper bound) and the shortage of production (lower bound), respectively. The superscripts o and g represent the production targets of ore and grade, and s is the number of realizations of orebody models. Given a discount rate r and period t, both the NPV and the unit costs of deviation decrease similarly by a ratio $1/(1+r)^t$ from the start-up of a mining project. The formulation of Equation 1.1 resembles the two-stage recourse model in stochastic optimization [6], where the mining decisions in the first state (part A) are not involved with uncertainty and the penalties of the deviations under geological uncertainty are used to correct the decisions in the second recourse stage (part B). As a result, two case studies of a gold deposit and a copper deposit applying the stochastic optimizer in Dimitrakopoulos and Ramazan [5] lead to increasing of NPV by 10% and 25%, respectively. Ramazan and Dimitrakopoulos [7] establish a more detailed stochastic integer programming (SIP) model of mine production scheduling, which considers the stockpile option. The application also shows the superiority of managing the deviations from production targets over the conventional optimizer and leads to higher NPV. The impact of geological uncertainty is even more far reaching when the mining workflows include more components to be optimized simultaneously, such as the operation of multiple mines, stockpiling, blending constraints and alternative processing streams throughout the supply chain from the mining operation to the end products in the market. Goodfellow and Dimitrakopoulos [8, 9] propose a new simultaneous stochastic optimization model that holistically optimizes the mine production schedule from extraction sequences to processing streams, while accounting for the geological uncertainty. Their experimental study in a copper-gold mine generates designs that have a good control on the risk of deviations from production targets and obtains an NPV that is 22.6% higher than an industry-standard mine planning software.

To summarize, the new paradigm of stochastic mine planning optimization increases the upside potential of mining projects and reduces the downside risks. Nevertheless, the potential value of managing risk under the framework of a stochastic mine planning optimizer can only be realized through the appropriate quantification of geological uncertainty. The stochastic simulation methods are by far the most practical approaches to quantify the geological uncertainty in orebody modelling [2, 10]. Since metal grades (tonnages) are the most sensitive factor in most mining projects [3] and are the common measurements in drilling core samples, new stochastic simulation methods with regard to metal grades are especially of interest in the present research, although the application of the methods should not be confined to mining area. In general, a random field is utilized to characterize the metal grades distribution in the three-dimensional space and the simulations are generated from probability distributions conditioned to the sample data with reproduction of spatial correlations of the random

field. Traditionally, the simulation methods assume that the random fields are dominated by multi-Gaussian probability distributions, and that the spatial correlations are characterized by two-point spatial statistics, such as covariance functions or variograms. Although the theory of traditional stochastic simulations is well established, as are their implementation and numerous applications from various area, there are some key limitations when applying the theory to orebody modelling in mine planning optimization. Firstly, the probability density functions of Gaussian distributions are symmetric, but metal grades usually have skewed probability distributions (with positive skewness in many cases). Thus, they coincide with non-Gaussian distributions instead. Secondly, two-point spatial statistics only deal with the statistical correlation between a pair of data in different locations and ignore the interactions among multiple data points. Thus, it cannot capture the spatial continuity, such as the connectivity of high values and low values. These drawbacks of two-point spatial statistics and the poor reproduction of complex spatial patterns have been reported in various publications [11–14]. Nevertheless, the spatial continuity of metal grades has a significant impact on the economic outcome of a mining sequence. For an intuitive understanding, this impact can be demonstrated by a simplified artificial example. As can be seen from Figure 1.1, the mine with more connectivity in the ore blocks has a higher revenue given the same distribution of economic values of blocks.

Figure 1.1 Revenues of extracting mining blocks with different spatial connectivity. The numbers represent the economic values of the blocks, and the blocks to be extracted are in red rectangles, assuming a slope constraint of 45°. The total revenue for the left mine is 12, but the revenue for the right one is 10.

As a remedy to deal with non-Gaussian probability in simulations, Gaussian anamorphosis [15] was proposed to transform the original data into normal scores before simulating and then reverting the transformation after simulating. Although the normal score transformation fulfils the prerequisites of traditional stochastic simulations, the preservation of the spatial statistics structure of the original data after reversion is not guaranteed. Moreover, Gaussian distribution has the maximum entropy among all the probability distributions with the same mean and variance [16]. Thus, simulations with Gaussian assumption have an inherent

weakness, as they do not capture the patterns with spatial structure, such as connectivity, other than the homogeneous randomness.

In one word, the framework of stochastic mine planning optimization, which can manage risks associated with geological uncertainty, relies on the quantification of uncertainty in orebody modelling. Therefore, new stochastic simulation methods for non-Gaussian random fields with the capacity to reproduce spatial continuity and complex spatial patterns should be developed as alternatives to the traditional ones. Recent research in high-order simulations reveals new concepts of spatial cumulants as a mathematical representation of multi-point statistics and is shown to be data-driven and competent in the reproduction of high-order spatial statistics [17–19]. Beyond the relatively few existing research, more refined theoretical models with high-order simulations and practical algorithms with effectivity and computational efficiency is worthy of further investigation. In addition, it is important to develop suitable software to launch some real-life applications of orebody modeling based on the new high-order simulation methods, and thus providing the quantification of geological uncertainty to advance the decision making in mining projects. These above reasoning contributes to the main motivations of the present research.

1.2 Research Goal and Objectives

The overall goal of the present research aims to develop new high-order stochastic simulation methods that can quantify spatial uncertainty of non-Gaussian random fields, and overcome the limitation of existing high-order simulation methods, particularly by improving the numerical stability with an approximation of conditional probability distribution by Legendre polynomial series. A new statistical learning framework will be established as the general foundation of the newly developed high-order simulation methods from which the random field models can be learned from different sources of data with the incorporation of the high-order spatial statistics. In addition, to meet the practical requirements from orebody modeling, the new high-order simulation methods will be extended to support training-image free simulation and address the difficulty of inferring high-order spatial statistics with relatively sparse sample data.

To achieve the above research goal, the research objectives are outlined as following:

1. A comprehensive review of stochastic simulation methods, especially focusing on the analysis of the limitation of existing high-order simulation methods, including the numerical instability issue of approximating a conditional probability density function (CPDF) and the computational efficiency of calculating the high-order spatial statis-

tics.

2. Develop a new computational model of approximating CPDF based on the concept of spatial Legendre moments allowing to accommodate flexible spatial templates and improving the computational efficiency of the approximation.

3. Develop a new high-order stochastic simulation method based on a statistical learning framework by representing the CPDFs in the functional space as spatial Legendre moment kernel Hilbert space, which not only improves the numerical stability of the previous high-order simulation methods but also owns the generalization capacity to mitigate the statistical conflicts between the sample data and the training image.

4. Develop a training-image free high-order simulation method that utilizes the interrelations between the low-order and high-order spatial statistics of the sample data, based on the new proposed kernel-based statistical learning framework.

5. Develop high-order simulation software to facilitate the generation of multiple realizations of ore reserve models with the reproduction of high-order spatial continuity to quantify the geological uncertainty and support stochastic mine planning optimization.

1.3 Thesis Outline

The thesis is organized into the following chapters:

- Chapter 1 provides a brief overview of the research background with a short explanation of the research motivation and main research objectives covered in the thesis.

- Chapter 2 presents a literature review of different kinds of stochastic simulation methods, including the traditional second-order geostatistical simulation methods, multiple-point simulation methods and high-order simulation methods, as well as their limitations.

- Chapter 3 introduces the general organization of the thesis and briefly explains the connections among the articles respectively corresponding to the main research objectives.

- Chapter 4 presents a new computational model of high-order stochastic simulation based on the concept of spatial Legendre moments, which significantly improves the computational efficiency and allows the variable spatial template during the simulation process.

- Chapter 5 proposes a new statistical learning framework in reproducing kernel Hilbert space to develop the high-order simulation method and improves the numerical stability of approximating the conditional probability density function by orthogonal polynomial expansion series. A solution to mitigate the statistical conflicts between the sample data and the training image is also provided.

- Chapter 6 further explores the relations between the low-order and high-order spatial statistics associated with a certain spatial template and from which a training-image free high-order simulation method is developed by utilizing the information from the so-called partially-matched replicates to the conditioning data.

- Chapter 7 develops and describes a software of high-order stochastic simulation based on statistical learning, incorporating the high-order spatial information at multiple scales from the sample data and the training image, which facilitates the generation of multiple realizations.

- Chapter 8 presents a general discussion of the methods developed in this thesis and their connections to the other methods.

- Chapter 9 concludes the thesis by highlighting the major contributions to high-order stochastic simulation methods and recommends related future work.

CHAPTER 2 LITERATURE REVIEW

Stochastic simulation aims to quantify the uncertainty of attributes in a random field by generating a set of realizations with equal probability of occurrence, whereas the original samples as the conditioning data remain unchanged. The reproduction of spatial statistics with respect to the real data is of the utmost importance in stochastic simulations. Through several decades of development, the quantification of spatial statistics has evolved from two-point spatial correlations to multi-point spatial continuity, or from low-order statistics to high-order statistics. Accordingly, stochastic simulation methods have also migrated from traditional variogram-based second-order simulation to multi-point simulation and high-order simulation, and furthermore from Gaussian to non-Gaussian random fields, which will be elaborated in the followed subsections. Other major stochastic simulation improvement made over the years are related to the implementation of various algorithms and their computational efficiency.

2.1 Sequential Simulation

In the 1970s, the turning-bands method [15, 20] was proposed to generate simulations that overcome the limitation of computational capacity, but this method has the shortcoming of loss of accuracy by approximation and struggled to reflect anisotropic covariance [21]. As an alternative, sequential simulation framework [22, 23] is adopted in most stochastic simulation methods up to now.

Consider a stationary and ergodic random field $\mathbf{Z}(\mathbf{u})$, let $Z(\mathbf{u}_1), \ldots, Z(\mathbf{u}_N)$ be a set of random variables with locations at $\mathbf{u}_1, \ldots, \mathbf{u}_N$, respectively. Then the N random variables $Z(\mathbf{u}_1), \ldots, Z(\mathbf{u}_N)$, constitute a joint multivariate distribution. In terms of stochastic simulation, suppose the realizations are to be generated from $Z(\mathbf{u}_1), \ldots, Z(\mathbf{u}_N)$, and the available data set are $\Lambda_0 = \{\zeta(u'_1), \ldots, \zeta(u'_n)\}$. For simplification,$Z(\mathbf{u}_1), \ldots, Z(\mathbf{u}_N)$ are alternatively written as $Z_1 \ldots, Z_N$, and similar simplification of notations entails knowing the context of a random field. Following the above notation, the stochastic simulation of the random field is based on the sampling from the N-variate probability distribution posterior to the data set Λ_0, which can be characterized by a conditional cumulative distribution function (CCDF) as $F_Z(z_1, \ldots, z_N|\Lambda_0)$ or by a probability density function (CPDF) as $f_Z(z_1, \ldots, z_N|\Lambda_0)$. The joint CPDF $f_Z(z_1, \ldots, z_N|\Lambda_0)$ can be decomposed into the product of a series of univariate

CPDFs [24, 25] as

$$f_Z(z_1, \ldots, z_N|\Lambda_0) = f_{Z_1}(z_1|\Lambda_0) \cdots f_{Z_N}(z_N|\Lambda_{N-1}), \tag{2.1}$$

where $\Lambda_i (i = 1, \ldots, N-1)$ are a series of sets with $\Lambda_i = \Lambda_{i-1} \cup \{\zeta(\mathbf{u}_i)\}, i = 1, \ldots, N$ and $\zeta(\mathbf{u}_i)$ are informed attribute values either from the sample or previous simulated values.

The basic idea of sequential simulation is to sequentially draw random values from the decomposed univariate CPDFs, following a random path to visit all the nodes to be simulated. Irrespective of the node's location corresponding to the sequence number, there is no difference in the sampling procedures. Without loss of generality, the CPDF in every single sampling procedure can be symbolized uniformly as, $f_{Z_0}(z_0|\Lambda)$, where Z_0 means the current simulating node and Λ means the set of conditioning data around Z_0's location $\mathbf{u}_0$. Considering the computational intensity and the statistical relevancy, the conditioning data are usually confined to a neighborhood closest to the simulation node instead of taking account of all available data on the whole domain of the random field. An algorithmic description of sequential simulation can be summarized as the following steps:

1. Draw a random path to visit all the N nodes to be simulated;
2. For each node $Z(\mathbf{u}_i)$, derive the conditional probability cumulative distribution $F_{Z_i}(z_i|\Lambda_{i-1})$ or the density function $f_{Z_i}(z_i|\Lambda_{i-1})$;
3. Draw a random value $\zeta(\mathbf{u}_i)$ from the conditional probability distribution in Step (2) and update the conditioning data by adding the node value $\zeta(\mathbf{u}_i)$ in to the current data set Λ_i;
4. Repeat from Step (2) until all the nodes are visited.

Since sequential simulation is flexible to accommodate various stochastic models and thus it becomes a mainstream way to implement different simulation methods, without further specification the stochastic simulation methods in the subsequent contexts are developed under this general framework.

2.2 Second-Order Stochastic Simulation

2.2.1 Sequential Gaussian simulation

Sequential Gaussian simulation (SGS) [22, 26, 27] is developed to generate random outputs from a Gaussian random field following the procedure of sequential simulation. An important

fact of SGS is that the conditional probability distribution of the variable to be simulated is a Gaussian distribution with mean as the simple kriging estimation and variance as the simple kriging variance. Thus, SGS can be implemented to successively draw random values from a sequential set of conditional probability distributions where the parameters are decided from solving simple kriging equations. Alternatively, the Gaussian random field can be decomposed as a mean field resulted from simple kriging and a residual random field respecting the covariance function from the data. In this way, Davis [21] proposed LU decomposition of the covariance matrix as a method to generate the simulated values simultaneously. Ideally, either SGS or simulation by LU decomposition assumes using the data from the whole domain. In practice, this assumption is always compromised by considering data within a certain neighbourhood due to intractability of the enormous matrix system with incorporation of all the nodes. This approximation is called screen effect approximation (SEA) since the closest data tend to screen the influence of farther data [26, 28]. Dimitrakopoulos and Luo [28] show that the accuracy of the above approximation can be measured by a function termed relative screen effect approximation (RSEA) loss, which depends on the ratio of posterior conditional variance with partial data and with full data. Furthermore, an algorithm named generalized sequential Gaussian simulation on group size ν (GSGS-ν), is developed to improve the computational efficiency of SGS dealing with large data [28]. GSGS-ν divides the N nodes on the simulation grid into a set of groups of size ν, and the nodes inside each group has the same neighborhood with conditional data up to a maximum number $\nu_{\max}$. A random path is selected to visit each group and simulation is conducted sequentially on each group using LU decomposition until all the nodes are visited. The computational cost of GSGS-ν is greatly reduced in comparison to SGS because the simulation of the nodes inside a group are not generated one by one through solving distinct simple kriging systems but were completed through one matrix decomposition. For the same sake of solving conditional simulation with large data, Vargas-Guzmán and Dimitrakopoulos [29] propose a new stochastic simulation method by successive residuals (CSSR). The CSSR method divides the covariance matrix to block matrices and develops a new approach to take LU decomposition in a stepwise way, and the mean of the conditional probability distribution is updated step by step by adding a successive set of residuals which is proven to be equivalent to normal simple kriging estimation [30]. Intuitively the procedure is to repeat filtering a mean component out of the current residual when new data is added and then a new residual remains after the filtering with the conditional covariance being updated at the same time; so each time adding the new data only the updated conditional covariance in the last step is needed to update the simulation result. Hence CSSR is able to solve conditional simulation of large scale and it also has the advantage to dynamically update the existed simulation results when new data are

incorporated without starting the whole simulation once again. Jewbali and Dimitrakopoulos [31] implement the CCSR method and apply to a stockwork gold deposit where additional infilled drill holes are used to update existed simulation results, however some extra storage of decomposed component of matrices are needed to facilitate the updating as a compromise of memory cost to gain computational efficiency.

2.2.2 Sequential indicator simulation

Sequential indicator simulation (SIS) [32–35] is a non-parametric simulation method which has no assumption of probability distribution, in contrary to the Gaussian assumption in SGS. To implement SIS, the original data are transformed to indicator code and the corresponding indicator random function is used to characterize the random field. The indicator function is defined as

$$i(u; z_k) = \begin{cases} 1 & z(\mathbf{u}) \leq z_k \\ 0 & \text{otherwise} \end{cases} \qquad k = 1, \ldots, K, \tag{2.2}$$

and z_k are the threshold values to divide the original data into $K + 1$ classes. The random function after indicator transformation of the original data can be written as

$$I(u; z_k) = \begin{cases} 1 & Z(\mathbf{u}) \leq z_k \\ 0 & \text{otherwise} \end{cases} \qquad k = 1, \ldots, K. \tag{2.3}$$

Then the conditional probability distribution of original random variable can be expressed as the expectation of the indicator random variable, that is

$$F[Z(\mathbf{u})|\Lambda] = \text{Prob}[Z(\mathbf{u}) \leq z_k|\Lambda] = E[I(\mathbf{u}; z_k)|\Lambda]. \tag{2.4}$$

The covariance functions or variograms are computed from the indicator data, and indicator kriging (IK) systems are solved to derive the conditional probability distribution (the expectation $E[I(\mathbf{u}; z_k)|\Lambda]$. Indicator kriging estimation is less sensitive to the outliers which is useful when the connectivity of high values are important [36]. In addition, it is possible to incorporate different sources of soft data into the simulation using SIS.

Since there are a certain number of categories in the SIS method, the cross-correlations can be derived experimentally from different pairs of random variables and cokriging can be used to estimate the expectation. In practice, alternative methods are indicator kriging on each categorical variable separately. In fact, Goovaerts [33] shows that there is no obvious advantage of using cokriging over the IK through a comparison study, on the contrary it may cause more problems of order deviations in SIS. The order deviation problems are linked

to the situation that the conditional probability distributions from SIS are not increasing functions as they should be, because there are no order relation constraints imposed on the kriging systems. One of post processing techniques to correct the probability distribution is based on downwards correction and upwards correction, where the lower part and higher part of the CDF curves are treated separately to ensure the non-decreasing characteristic and then take the average between them as the corrected CDF [26]. Another practical issue with SIS is that the estimation near the lowest and highest cut-offs maybe unreliable because the scarce of data with increasing number of categories, and thus the tails of CDF often are extrapolated, for instance, by linear functions, hyperbolic functions, or power functions [26]. With appropriate simplification in modelling the cross-correlations and post processing on CDF, SIS method is a powerful tool to generate realizations for non-Gaussian random field, however, the limitation of two-point geostatistical framework restricts its application to circumstances with relatively simple spatial structures.

2.2.3 Extension to multiple variables

It is very common in earth science data that various geological attributes are correlated in addition to spatial correlations of their own. Furthermore, the interrelation between multiple variables are also dependent on their spatial configurations, adding the complexity to model the random fields. The analysis of covariance or variograms of multivariate random fields is called coregionalization analysis in geostatistics [37]. The linear model of coregionalization (LMC) is the most widely used model applying to coregionalized variables for its simplicity [26, 38, 39]. Instead of being a single random variable, there is a random vector corresponding to each location in the multivariate random fields. To make the difference in notation, here $\underline{\mathbf{Z}}(\mathbf{u}) = \big(\underline{Z}_1(\mathbf{u}), \ldots, \underline{Z}_n(\mathbf{u})\big)$ is used to represent the multivariate random field with n distinct attributes. LMC assumes each random variable $\underline{Z}_i(\mathbf{u})$ can be decomposed into a linear combination of $L+1$ basic structures consisting of independent components with a unit covariance function, that is,

$$\underline{Z}_i(\mathbf{u}) = \sum_{l=0}^{L} \sum_{k=1}^{n_l} a_{ik}^l Y_k^l(\mathbf{u}) + m_i, \tag{2.5}$$

where $m_i = E[\underline{Z}_i(\mathbf{u})]$ and $Y_k^l(\mathbf{u})$ are the unit random components with zero mean and n_l is the number of independent components in each basic structure, while $Y_k^0(\mathbf{u})$ corresponds to the component with a nugget effect. As $Y_k^l(\mathbf{u})$ are independent, their correlations can be

expressed as

$$cov[Y_k^l(\mathbf{u}), Y_{k'}^{l'}(\mathbf{u}+\mathbf{h})] = \begin{cases} c_l(\mathbf{h}) & \text{if } k = k', l = l' \\ 0 & \text{otherwise} \end{cases}. \tag{2.6}$$

The cross-covariance can be expressed as

$$C_{ij}(\mathbf{h}) = \sum_{l=0}^{L} \sum_{k=1}^{n_l} a_{ik}^l a_{jk}^l c_l(\mathbf{h}), \tag{2.7}$$

or in the matrix form as

$$C(\mathbf{h}) = \sum_{l=0}^{L} B_l c_l(\mathbf{h}), \tag{2.8}$$

where B_l is an $n \times n$ matrix given the number of variables n with $B_l = \left[\sum_{k=1}^{n_l} a_{ik}^l a_{jk}^l\right]_{n\times n}$. The cross-variogram matrix $\Gamma(\mathbf{h})$ can be derived similarly as

$$\Gamma(\mathbf{h}) = \sum_{l=0}^{L} B_l \gamma_l(\mathbf{h}), \tag{2.9}$$

where $\gamma_l(\mathbf{h})$ are the unit variograms of the basic components.

Since there are in total n direct variograms and $n(n+1)/2$ cross-variograms to be modeled given n random variables, the computation of modeling the spatial correlations is tedious and leads to large cokriging systems to solve. Practically the multiple variables are often decomposed into decorrelated new random variables by taking linear transformation and then reverse back to the original data space by taking back transformations on separate simulations of each single decorrelated variable. The principal component analysis (PCA) has been used to decorrelate the coregionalization variables in the past [40–42], however, the decorrelation with PCA is only guaranteed in zero-lag variance-covariance matrices yet ignoring the spatial correlations within the data. An alternative approach transforms the corregionalized variables into so called Minimum/Maximum autocorrelation factors [43], which is proven to be decorrelated on all the lags provided there are no more than two structures in LMC. In this case, the cross-variogram matrix $\Gamma(\mathbf{h})$ is written as

$$\Gamma(\mathbf{h}) = B_1\gamma_1(\mathbf{h}) + (B - B_1)\gamma_2(\mathbf{h}), \tag{2.10}$$

where B is the variance-covariance matrix and B_1 is the cross-covariance matrix for the first structure. The spectral decomposition on B gives $B = Q\Lambda Q^T$ with Q be the orthogonal matrix of eigenvectors. Thus, the PCA factors with respect to B can be written as

$$\mathbf{Y}(\mathbf{u}) = \Lambda^{-\frac{1}{2}} Q^T Z(\mathbf{u}) = AZ(\mathbf{u}). \tag{2.11}$$

Take variance-covariance analysis on $\mathbf{Y}(\mathbf{u})$ one can get

$$A\Gamma(\mathbf{h})A^T = I\gamma_2(\mathbf{h}) + AB_1A^T[\gamma_1(\mathbf{h}) - \gamma_2(\mathbf{h})]. \tag{2.12}$$

Another spectral decomposition on AB_1A^T gives $AB_1A^T = Q_1\Lambda_1Q_1^T$ with Q_1 as the orthogonal matrix of eigenvectors. The MAF factors are expressed as

$$\mathbf{F}(\mathbf{u}) = Q_1^T\mathbf{Y}(\mathbf{u}) = Q_1^T\Lambda^{-\frac{1}{2}}Q^T\mathbf{Z}(\mathbf{u}). \tag{2.13}$$

Therefore, the variance-covariance analysis on $\mathbf{F}(\mathbf{u})$ gives

$$\Gamma_{\mathbf{F}}(\mathbf{h}) = \Lambda_1\gamma_1(\mathbf{h}) + (I - \Lambda_1)\gamma_2(\mathbf{h}) \tag{2.14}$$

Thus, the MAF factors $\mathbf{F}(\mathbf{u})$ are decorrelated on all lags since the corregionalization matrices are diagonal. Desbarats and Dimitrakopoulos [44] develop the co-simulation method base on the MAF transformation. The authors perform simulations based on MAF factors with suitable univariate simulation method and then the results are back transformed to the original data space to obtain realizations with reproduction of cross correlations as well as spatial correlations. For the teaching aid of joint simulation with MAF as well as detailed explanation of application, the readers are referred to Rondon [45].

2.2.4 Simulation on block support

Extension of stochastic simulation to block support is important in mining applications since the orebody models are frequently represented by blocks with volumes comparable to selective mining units (SMU). In terms of available data in mine planning, the block simulation becomes complicated problem since there are various information in different scales of support, including core samples, mined blocks, stopes, and bulk samples [46]. The naive way to deal with changing of support is to discretize the blocks into point supports with smaller volumes and then take the average as the data value of the blocks. However, this is practically infeasible for orebody modeling of large mines with millions of blocks.

Marcotte [46] proposes a method to generate realizations on block support based on the model of disjunctive kriging, where the data on various supports are expressed as the normalized Hermite polynomial series after Gaussian transformation. The coefficients of Hermite polynomial series with respect to the data in block support can be derived through computing an integral related to the Hermite polynomials on point support. However, the computation is complex and the method also relies on an assumption that the random variables in point

support and block support follow a bivariate Hermite distribution. Emery [47, 48] proposes an alternative block simulation method based on discrete Gaussian model where an explicit function related to the so-called change-of-support coefficient, is established to model the cross-variograms between random variables at point support and block support. As an extension to GSGS-ν, direct block simulation (DBSIM) has been developed to simulate large orebodies [49]. The block size can be the same as the selective mining units, and the blocks are subdivided into a group of internal nodes. The simulation is carried out on these internal nodes through LU decomposition, which is similar in the way to GSGS-ν, however, the values of internal nodes are discarded after the value of block is set as the average of the internal nodes and the block values are directly included as the new conditioning data in the followed simulations. This algorithm turns out to be fast because it significantly reduces the scale of simulation problem after changing of support from points to blocks. Due to the change of support in DBSIM, dilution effect may exist as a result of smoothing average and also the covariance or variograms should be calculated separately between points to points, points to blocks, and blocks to blocks, respectively. Several successful applications to large mine are available [50, 51], and the simulation method is also extended to co-simulation with MAF transformation by Boucher and Dimitrakopoulos [52]. All the above methods either assume a Gaussian distribution or need Gaussian transformation of the random variables, hence have the limitation to simulate non-Gaussian random fields.

2.3 Multi-Point Simulation

Multi-point simulation (MPS) allows to reproduce spatial structures with multi-point interactions, for example, the connectivity of extreme values of metal grades, or curvilinear channels in reservoir models. In contrast to the two-point covariance or variograms which only consider the pairwise second-order statistics, the multi-point statistics involve multiple random variables with regard to the random field and thus lead to high-order statistics. Generally, multi-point statistics do not rely on the Gaussian model and encompass far more spatial structures than two-point statistics which works perfectly for Gaussian model. It turns out that two random fields with distinct spatial structures, say continuous channels and lens structures, could share the similar variograms [53, 54], which clearly indicates the limitation of second-order statistics to capture the complex spatial patterns (Figure 2.1).

In contrast to the second-order spatial statistics being represented by either the covariance or the variogram, the multi-point statistics are usually borrowed from an exhaustive training image (TI). Several key concepts from MPS are as follows:

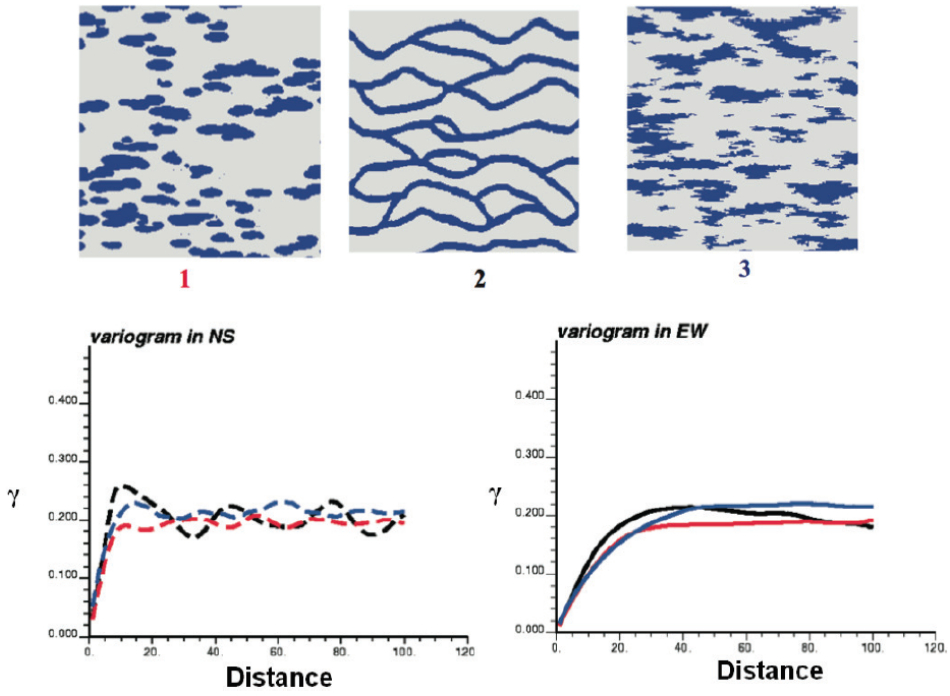


Figure 2.1 Limitation of second-order statistics to characterize spatial patterns (From [53])

- **Training image**: An exhaustive image assumed to share similar spatial structures with the attributes to be simulated. The training image can also be regarded as a representative realization of the prior random function model before the actual data are incorporated to update the posterior distribution of the random field [55].
- **Spatial template**: The multi-point statistics are defined on multiple random variables from a random field. The geometric configuration associated to the multiple random variables is termed the spatial template. Usually the spatial template can be denoted by a set of distance vectors apart from the center node to be simulated as $\mathbf{T} = \{\mathbf{h}_1, \ldots, \mathbf{h}_n\}$.
- **Data event**: The conditioning data values retrieved from either the samples or the simulated nodes within the spatial template is called a data event.

Depends on the implementation, the MPS methods falls into two categories of simulations [56]:

- **Pixel-based MPS**: The MPS methods where the simulations are carried out sequentially pixel by pixel are called pixel-based algorithms.
- **Pattern-based MPS**: The simulations are generated as patches instead of as a single value, sometimes it is also called as patch-based MPS methods.

2.3.1 Pixel-based MPS

2.3.1.1 SNESIM

The first popularized MPS algorithm, the so called single normal equation simulation algorithm (SNESIM), is proposed by Strebelle [57]. The SNESIM algorithm serves as an extension to the extend normal equations simulation (ENESIM) algorithm proposed by Guardiano and Srivastava [12], solving the computational difficulty in ENESIM and leading to practical use of MPS. Suppose the node to be simulated as $Z(\mathbf{u}_0)$ and let the template be denoted as $\boldsymbol{\tau}$ with the data event associated to the it as $d_n = \{Z(\mathbf{u}_1) = z_1, \ldots, Z(\mathbf{u}_n) = z_n\}$ where $z_i \in \{1, \ldots, K\}$ belongs to one of the K categories. Then the indicator variance/covariance between the $Z(\mathbf{u}_0)$ and data event d_n can be derived as

$$Cov\Big[I(\mathbf{u}; z_k), I(\boldsymbol{\tau}; d_n)\Big] = P\Big[Z_0(\mathbf{u}) = z_k, d_n\Big] - P\Big[Z_0(\mathbf{u}) = z_k\Big] \cdot P(d_n), \tag{2.15}$$

$$Var\Big[I(\boldsymbol{\tau}, d_n)\Big] = P(d_n) \cdot \Big[1 - P(d_n)\Big]. \tag{2.16}$$

Thus, the simple kriging equation can be solved by a single normal equation as

$$\lambda_{sk} = \frac{Cov\Big[I(\mathbf{u}; z_k), I(\boldsymbol{\tau}; d_n)\Big]}{Var\Big[I(\boldsymbol{\tau}, d_n)\Big]}. \tag{2.17}$$

With simple substitution, the conditional probability can be obtained as

$$P\Big[Z_0(\mathbf{u}) = z_k | d_n\Big] = E\Big[I(\mathbf{u}; z_k) | d_n\Big] = \frac{P\Big[Z_0(\mathbf{u}) = z_k, d_n\Big]}{P(d_n)}, \tag{2.18}$$

which provides the equivalency of simple kriging solution and Bayes' equation.

From Equation (2.18), the conditional distribution can be estimated from the frequency of each category among the replicates of data event in the TI. However, the brute-force searching for the data event from the TI is time consuming and it is the reason that ENESIM algorithm remains only of theoretical interest. Instead of searching anew the replicates of the data event for each node to be simulated, a tree data structure is used in SNESIM to store the possible replicates in the TI associated to the geometry template. Therefore, only one run of scanning the TI is needed and the searching time for replicates of data events is greatly reduced by traversing a tree with depth comparable to the number of nodes in the template. In case that the number of exact replicates of data event d_n is not enough to make reasonable estimation, the most distant node from the center node is dropped with replacement of d_n as its subset d_{n-1} and this procedure repeats until the number of replicates reaches a minimum threshold.

The marginal distribution is used to draw a random value if none subset of the data event corresponds to the replicates in TI with statistical significance.

2.3.1.2 IMPALA

Although SNESIM is fast in computation, it is memory demanding because the number of tree nodes increases very fast when the template size or the number of category increases, which eventually increases the searching time as well. Instead of using a tree data structure, Straubhaar et al. [58] propose an improved parallel multi-point simulation algorithm (IMPALA) where a list data structure is used to store the replicates from the training image. Similar to SNESIM, a searching template is used to scan the TI for only once to construct the list, but only the leaf nodes are stored in comparison to the tree structure in SNESIM, and hence the memory usage is greatly reduced so that a large template may be applicable in the algorithm. However, the searching is more CPU demanding than SNESIM due to the serial feature of the list data structure. A list sorted by the number of occurrences of a certain category in the reference node is introduced to accelerate the searching, and furthermore, the list data structure also enables the parallelized implementations which have more computational efficiency [58, 59].

2.3.1.3 Direct sampling

Rather than scanning the whole training image to explicitly build the conditional distribution, Mariethoz et al. [60] propose a new direct sampling (DS) method that allows to draw a random value directly from the scanning procedure. The main steps of DS are as the following:

(1) Once a certain data event d_n was given, the algorithm starts to randomly pick a replicate d'_n from the training image with the same template associated to the d_n, and a distance function is defined to measure the similarity between d_n and d'_n.

(2) If the distance between d_n and d'_n is less than a certain threshold, i.e., they are deemed as similar events. Then the value in the reference node of d'_n is set as the data value of the node to be simulated and repeat from Step (1).

(3) Otherwise, store the minimal distance between d_n and the replicate up to the current scanning on the TI, together with the corresponding reference node value.

(4) If no satisfied replicate is found up to some specified maximum times of scanning, then chose the reference node value from the replicate which is most similar to the data event and repeat from Step (1).

In contrast to SNESIM, DS algorithm requires no additional memory usage and the similarity measure is defined to find replicates instead of searching for exact replicate by dropping some nodes. The running time may depend on the conditional probability distribution since the first occurrence of the replicate of a certain data event varied regarding the distribution, however an upper bound can be derived from the maximum threshold for scanning. In addition, it is also possible to simulate continuous random fields once the related distance function is defined. For the practical guide to apply DS algorithm, the reader is referred to Meerschman et al. [61].

2.3.1.4 Computationally improved methods of pixel-based MPS

More recently, Strebelle and Cavelius [62] investigate the main factors affected the memory and CPU usage including the size of the template and the proportion of informed nodes in the data event. They propose a new multiple grid method where extra intermediary sub-grids are included to increase the proportion of the informed nodes in the data event. Moreover, the data template to preferentially selecting simulated nodes is suggested to include more informed data in a relatively small template and an optimal choice of template size is also introduced. These techniques are combined in a new version of SNESIM to solve the memory and speed issues. Several of the above-mentioned pixel-based MPS algorithms are also modified to be implemented with parallelized computations on the graphical processing units (GPU) and are reported to speed up simulation within orders of tens to hundreds [63, 64]. However, it should be noted that GPU-version algorithms are hardware demanding and the GPU memory could be a limitation of conducting large scale simulations.

2.3.2 Pattern-based MPS

Despite the quick developments of pixel-based algorithms, their limitations are that the exact replicates of data events are not easy to find, and in addition, most of these algorithms only apply to the categorical data. The reason is that the statistical model with regard to pixel-based MPS methods is essentially a discrete distribution relied on the counting number of data events occurring in the training image, which becomes unstable when replicates are few. These limitations are overcome by the pattern-based MPS methods as followed.

2.3.2.1 SIMPAT

Arpat [65] proposes a new stochastic simulation algorithm with patterns (SIMPAT) which abandons the ideology of pixel-based simulations within a specified statistical framework.

Instead of drawing a random value for each single node to be simulated, SIMPAT is developed to search a similar pattern to the data event from the training image to be embedded to the simulation grid as a whole, and the simulation algorithms followed in this paradigm are categorized as pattern-based or patch-based MPS algorithms.

Suppose that the template $\boldsymbol{\tau}$ is defined by a set of distance vectors as $\boldsymbol{\tau} = \{\mathbf{h}_1, \ldots, \boldsymbol{h}_T\}$, then the data event on the simulation grid **re** is defined as $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau}) = \{\mathbf{re}(\mathbf{u} + \mathbf{h}_1), \ldots, \mathbf{re}(\mathbf{u} + \mathbf{h}_T)\}$ where $\mathbf{u}_0$ represents the node to be simulated, and note that some nodes may be uninformed in the data event. For a specified node at location $\mathbf{u}$ on the training image **ti**, a pattern within the template can be defined as $\mathbf{pat}(\mathbf{u}; \boldsymbol{\tau}) = \{\mathbf{ti}(\mathbf{u}), \mathbf{ti}(\mathbf{u}+\mathbf{h}_1), \ldots, \mathbf{ti}(\mathbf{u}+\mathbf{h}_T)\}$. A distance function can be defined to measure the similarity of the data event $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})$ and the pattern $\mathbf{pat}(\mathbf{u}; \boldsymbol{\tau})$ form the training image. For instance, Manhattan distance was used in SIMPAT as

$$d\Big(\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau}), \mathbf{pat}(\mathbf{u}; \boldsymbol{\tau})\Big) = \sum_{\alpha=1}^{T} |\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})(\mathbf{h}_\alpha) - \mathbf{pat}(\mathbf{u}; \boldsymbol{\tau})(\mathbf{h}_\alpha)|, \tag{2.19}$$

which means the more similar are the data event $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})$ and the pattern $\mathbf{pat}(\mathbf{u}; \boldsymbol{\tau})$ as the distance becomes smaller. The computation of Manhattan distances will be skipped from the uninformed nodes in the data event. The main procedures of SIMPAT are quite straightforward as the following:

(1) For each node $Z(\mathbf{u}_0)$ to be simulated, search the neighborhood within the template $\boldsymbol{\tau}$ to include the sample data or previously simulated nodes into the data event $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})$.

(2) Scanning the training image to find the pattern $\mathbf{pat}^*(\mathbf{u}; \boldsymbol{\tau})$ that is the most similar to $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})$.

(3) Assign the data values of the most similar pattern $\mathbf{pat}^*(\mathbf{u}; \boldsymbol{\tau})$ to all the nodes inside the template around the reference node $\mathbf{u}_0$. If there are more than one patterns that have the similarity measure to $\mathbf{dev}(\mathbf{u}_0; \boldsymbol{\tau})$, then randomly pick one of them as the representative.

(4) Repeat Step (1) until all the nodes on the simulation grid are simulated.

2.3.2.2 FILTERSIM

Although the similarity measure is more flexible than searching the exact replicates, the computation of distances throughout the entire pattern database for each node to be simulated is CPU demanding. To alleviate the computational cost of similarity comparison, Zhang et al. [66] propose a simulation method using filter scores (FILTERSIM) which classifies the patterns into different categories by applying certain filter functions.

For a training image in 2D space, let $X(i,j)$ be the datum value at location (i,j), and let the dimension of the template be $(2n+1) \times (2n+1)$, then the filter score associated to a specified filter function $f(u,v)$ can be defined as the following [66]:

$$S_f(i,j) = \sum_{v=-n}^{n} \sum_{u=-n}^{n} f(u,v) \cdot X(i+u, j+v). \tag{2.20}$$

In 2D space, there are six filter functions defined in FILTERSIM algorithm. Each filter gives different weights to the nodes on the template to capture different spatial features in the pattern. For instance, the sixth filter is defined in FILTERSIM as $f_6(u,v) = \frac{2|v|}{n} - 1$, which visually appears like Figure 2.2 capturing the E-W curvature of the pattern. Thus, for each pattern on the 2D training image, there are in total six scores assigned to the pattern which transform the pattern of size $(2n+1) \times (2n+1)$ into a point in six-dimensional filter score space. The filter scores for each of the 6 filters are further divided into 5 segments by their quintile thresholds, and hence the whole filter score space is discretized into 5^6 classes where each pattern falls into one of the classes. In practice, the actual number of classes would be much less than 5^6 because many classes are empty due to the limited number of patterns encountered in the training image. Each class then contains various number of patterns and a training prototype is defined as the average pattern in this class. The rest of the simulation is as follows:

(1) For each node $Z(\mathbf{u}_0)$ to be simulated, retrieve the data event associated to the template. Compare the similarity between the data event and all the prototypes to find the class with the patterns closest to the data event, by a predefined distance function.

(2) Randomly draw a pattern from the candidate class and paste it to the location centered at $\mathbf{u}_0$.

(3) Repeat from Step (1) until all the nodes are simulated. In addition, a fixed inner part is defined within the template for each pattern. As a rule, the hard data and the inner part of the simulated nodes within a pasted pattern are frozen during the next sequences of simulations.

As one time scanning throughout the training image is needed to build the filter score space and after that the similarity comparison only takes place between the data events and prototypes, thus FILTERSIM is quite computationally efficient. However, the simple filter functions may not be able to classify complex spatial patterns effectively and tend to deteriorate the reproduction of spatial continuity.

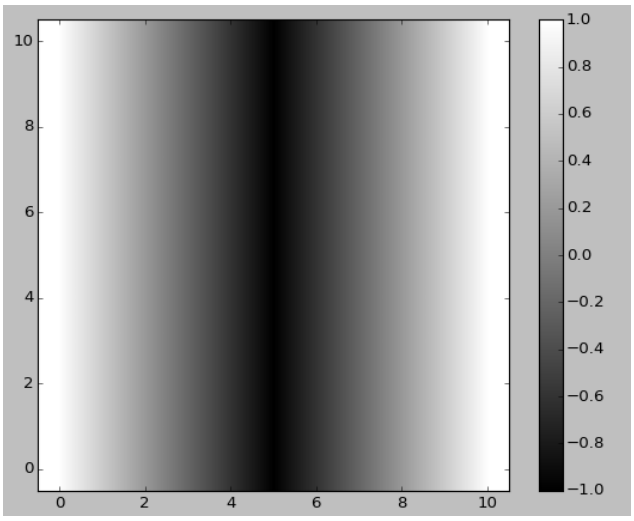


Figure 2.2 The sixth filter defined in FILTERSIM as a reflection of E-W curvature

2.3.2.3 DISPAT

Honarkhah and Caers [67] propose a more advanced approach to classify the patterns which leads to a new MPS algorithm with distance-based modelling of patterns (DISPAT). The first step of the above algorithm is to obtain a similarity matrix between all the patterns by computations on some predefined distance function. Then multidimensional scaling (MDS) technique is applied to the similarity matrix to reduce the patterns to points in lower dimensional space while preserving the order relations of the distances in the original space. The patterns embedded in the new metric space after MDS transformation are classified into a certain number of classes using the kernel k-means algorithm [68] where the points are actually again mapped to a kernel feature space. The rest part of the algorithm is similar to other pattern-based algorithms to compare the data events with the prototype of each class and randomly draw a pattern from the most similar class. Although the new algorithm has better reproduction of spatial continuity than FILTERSIM, however, the implementation is also more complicated , and several parameters including the size of dimension in MDS algorithm and the number of classes in k-means algorithm need to be deliberately selected.

2.3.2.4 WAVESIM

Chatterjee et al. [69] propose a pattern-based simulation algorithm using wavelet analysis (WAVESIM). Wavelets are defined as a family of bases by translation and dilation of a square

integrable function, namely as the mother wavelet, which can be written as

$$\psi_{j,k}(x) = 2^{-\frac{j}{2}}\psi(2^{-j}x - k), \quad j,k \in \mathbb{N}, \tag{2.21}$$

where j, k are the scaling parameter and translation parameter, respectively; $\psi_{j,k}$ and ψ are the wavelets and mother wavelet, respectively. Scaling bases $\phi_{j,k}(x)$ are defined in the similar way by a scaling function $\phi(x)$ as

$$\phi_{j,k}(x) = 2^{-\frac{j}{2}}\phi(2^{-j}x - k), \quad j,k \in \mathbb{N}. \tag{2.22}$$

A square integrable function $f \in L^2(\mathbb{R})$ can be reconstructed from the wavelets and scaling functions as

$$\hat{f}(x) = \sum_{k=0}^{N_J-1} a_{J,k}\phi_{J,k}(x) + \sum_{j=1}^{J}\sum_{k=0}^{N_j-1} w_{j,k}\psi_{j,k}(x), \tag{2.23}$$

where $\hat{f}(x)$ is the reconstructed function and J is the levels of wavelet decomposition; $N_j = \frac{N}{2^j}$ and N is the original range of the domain. For a 2D image, the discrete wavelet composition (DWT) leads to one approximate sub-band image as the scaling image and three high-frequency sub-band images as wavelet images. As can be seen from Equation (2.23), these sub-band images have size $1/2^j$ of the original image in the j-th level of decomposition. The approximate sub-band image is used as the representation of a pattern after DWT at a specified scale J. Thus, the patterns are reduced in dimension by ratio of $1/2^J$, and the approximated sub-band images are further classified into different classes using k-means algorithm. Prototypes are generated for each class of pattern afterwards and the simulation is carried out similarly as other pattern-based methods. A slight difference is that a distribution of patterns inside a specified class is estimated from the empirical distribution of the center node and the simulated patterns are randomly drawn according to this probability distribution. More recently, Chatterjee et al. [70] develop an updated version of WAVESIM which conducts pattern-based simulation on images in wavelet domain and take the inverse discrete wavelet decomposition (IDWT) to generate realizations in the spatial domain.

2.3.2.5 CDFSIM

Mustapha et al. [71] propose a simulation algorithm through decomposition of cumulative distribution functions of transformed spatial patterns (CDFSIM) which transforms the patterns into one-dimensional real data by a non-linear function. The transformation function

is defined as

$$f\Big(\mathbf{ti}(\mathbf{u};\boldsymbol{\tau})\Big) = \sum_{j=1}^{T} \mathbf{ti}(\mathbf{u}+\mathbf{h}_j)s(j), \tag{2.24}$$

$$s(x) = \begin{cases} 1 & \text{for continuous image} \\ \dfrac{1}{x} & \text{for binary image} \end{cases}. \tag{2.25}$$

Mustapha et al. [71] construct a cumulative distribution function from the transformed data and the corresponding location indices, and the patterns are classified into a certain number of classes by an algorithm based on the decomposition of thresholds from the above 1D distribution. This algorithm is computationally efficient and easy to implement. It is proven by Mustapha et al. [71] that if the distance between two transformation functions is bigger than r, then the two corresponding patterns will also have a distance larger than r. However, the converse of the claim is not necessarily to be true, thus the transformation does not fully preserve the distance order, which may lead to discontinuity in the simulations.

2.3.2.6 Pattern-based MPS as extensions from other methods

As an extension to pixel-based DS algorithm, Rezaee et al. [72] implement a pattern-based version of DS where the replicates from the TI resembling the data event are pasted to the simulation grid instead of a single node. Noting that boundary continuity may not be maintained in common pattern-based simulations, there are several methods borrowing the concepts of texture synthesis to MPS algorithm which alleviate the boundary conflicts [73–75]. These methods generally use a so called unilateral raster path to generate realizations instead of a random path, which means that the simulation is growing along some fixed directions. The unilateral path is proposed in texture synthesis to reproduce patterns from a reference texture, assuming a Markovian property [76]. In compliance with the Markovian property, the nodes are visited in a regular path and only precedent nodes can be included as conditional data for the current node to be simulated. Parra and Ortiz [75] abandon the Markovian assumption and adapt the texture synthesis to conditional simulation in two steps running. The causal nodes in the template which comply to the Markovian property are firstly used to search candidate replicates from the training image, and then in the second run the non-causal nodes are added to refine the final searching. Tahmasebi et al. [77] also adopt the raster path in their implementation of a cross-correlation based simulation (CCSIM). The cross-correlation function is used as the distance measure between patterns in replacement of the Euclidean distance, which is proven to be more computationally efficient. Furthermore, CCSIM introduces an overlapping region between the pasted patterns and

the current simulation area within the template. And the cross-correlation between the overlapping region and the TI is computed as the similarity measure, in this way the boundary continuity can be well reproduced with less computational time. Mahmud et al. [74] propose a conditional image quilting algorithm (CIQ) which is implemented similarly to CCSIM, however, CIQ enhances the boundary continuity by a dynamic programming algorithm named image quilting [78]. For a thorough review of relating texture synthesis to MPS, the readers are referred to Mariethoz and Lefebvre [79].

2.3.2.7 Multiple grid simulation

For computational reason, the template size in MPS algorithms should be relatively small compared to the size of TI, however a negative impact is that the large spatial structure may not be reflected from the template. The idea of multiple grid simulation [80] is utilized in various MPS algorithms to solve the problem. The simulation grid D is divided into multiple nested grids $D_g(1 \leq g \leq G)$ with every 2^{g-1}-th node picked from the original grid $D(g = 1)$, and correspondingly a series of template $\boldsymbol{\tau}_g(1 \leq g \leq G)$ are generated in the same way from the original template $\boldsymbol{\tau}(g = 1)$. The simulations are carried out sequentially from the coarsest grid D_G with the template $\boldsymbol{\tau}_g$ of the largest size to the finest grid $D(g = 1)$ with the template $\boldsymbol{\tau}(g = 1)$. Figure 2.3 shows the relation between the coarse grid and the fine grid with cascading templates, and Figure 2.4 shows how the coarse template is used to retrieve data. The strategy of multiple grid simulation is able to integrate spatial structures at different scales and actually it has been applied to either pixel-based MPS or pattern-based MPS algorithms, although the implementation may have some difference.

2.3.2.8 Scope of application to MPS

In general, pixel-based MPS algorithms are broadly used to reproduce spatial continuity of categorical random field within the Bayesian framework. Their computational efficiency varies in terms of different implementations or data structures adopted. However, the practical difficulty to find exact replicates of data events is often a hindrance to reproduce the multi-point interactions. By contrast, pattern-based MPS algorithms are more flexible to draw patterns from the TI by introducing the concept of similarity measure with distance functions. Thus, pattern-based MPS algorithms can be applied to both categorical and continuous data with reasonable reproduction of spatial structures. The similarity measure between patterns in fact replaces the role of the conditional probability distribution. The cutting down of statistical model in pattern-based MPS algorithms is twofold. On the one hand, it eases the computational burden without estimation of a conditional probability dis-

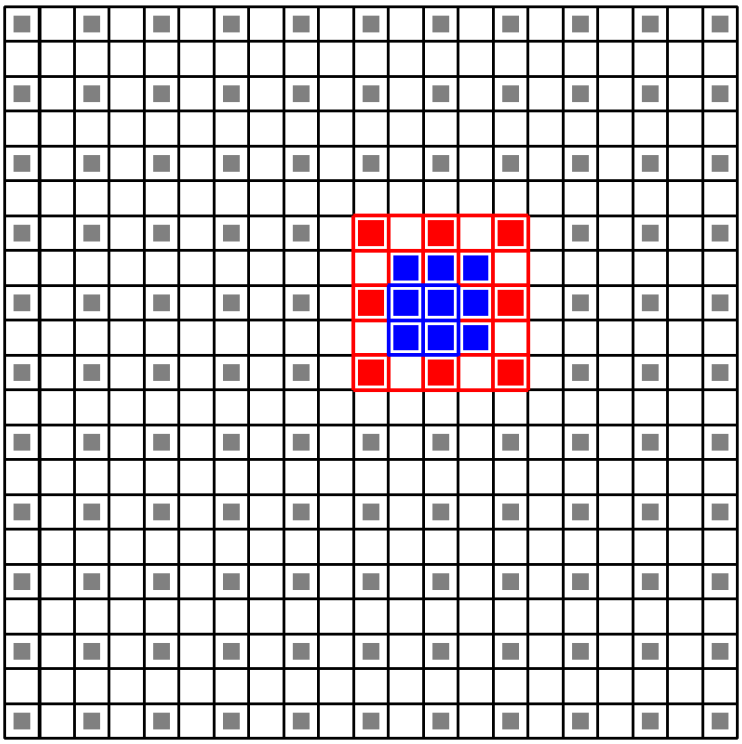


Figure 2.3 Coarse grid (grey nodes) with the coarse template (red nodes) overlapping on the fine grid with fine template (blue nodes); note that the coarse template has bigger size to capture larger spatial structures.

tribution. On the other hand, it also means less strictness in mathematical sense, which may limit their reliability in practical applications. Most importantly, either pixel-based or pattern-based MPS algorithms are eventually training image driven, hence their performance depends on the quality of the training image, and it is hard to resolve the possible conflicts between the training image and the hard data.

It should be noted that there are also other simulation methods intending to simulate complex spatial features and of which the framework is out of the scope of MPS. For instance, the object-based simulation algorithms have been developed to use parametric shapes as the basic simulating units to fit the conditioning statistics through an iterate process [81–83]. However, the object-based simulation algorithms are limited in the capability to respect conditioning data [84], and instead they can act as alternative methods to generate training images for running MPS [85].

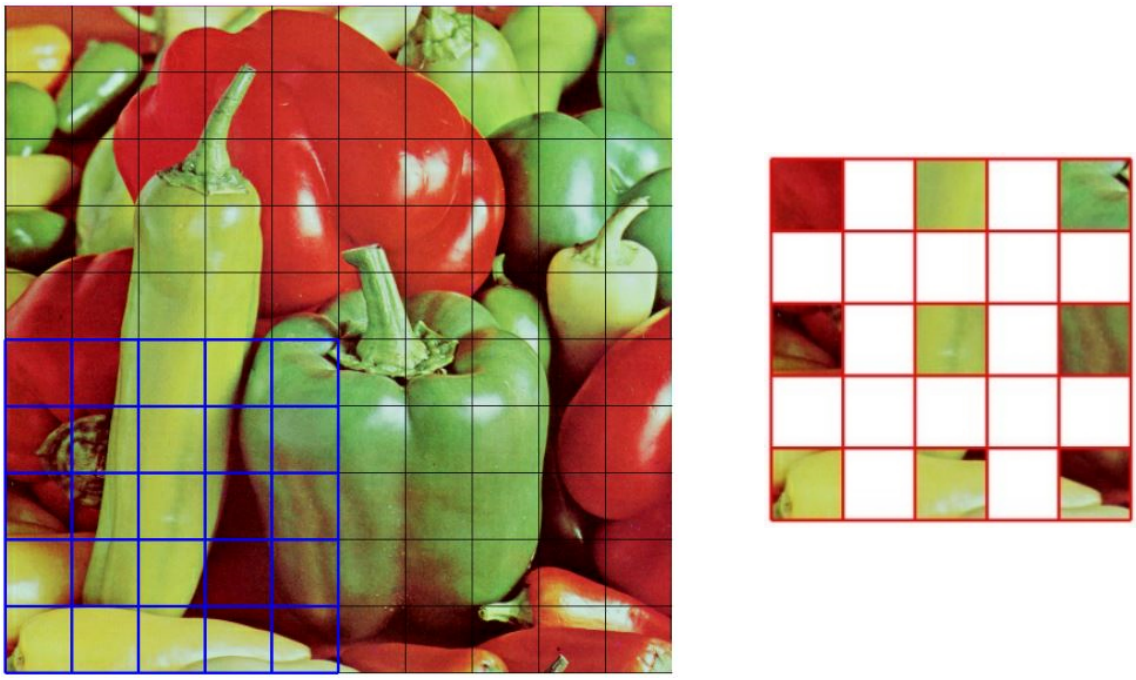


Figure 2.4 A replicate (right) retrieved by a coarse template form the training image (left). (Background image from `http://sipi.usc.edu/database/`)

2.4 High-Order Stochastic Simulation

High-order stochastic simulation methods aim to generate realizations of non-Gaussian random field without presumptions of probability distribution and to reproduce high-order spatial statistics among multiple points, as a new approach to overcome the limitations of traditional second-order stochastic simulations. The terminology of multi-point statistics and high-order statistics are synonyms in many existing literature. However, to be brief, the high-order simulation methods differ from the mainstream MPS algorithms in two aspects. On the one hand, the formulation of statistical models in high-order simulation is different from the pixel-based MPS algorithms which attribute to solving a single normal equation. On the other hand, the difference from the patter-based MPS methods is obvious since there is no explicit statistical model established by the patter-based MPS methods.

2.4.1 Spatial cumulants

Starting from an effective quantification of high-order statistics, Dimitrakopoulos et al. [17] propose a new concept of spatial cumulants, and upon which a statistical model of approximating the multivariate CPDF by Legendre polynomial expansion series is established for stochastic simulation in Mustapha and Dimitrakopoulos [18]. Cumulant generating function

is defined as the logarithm of moment generating function, and cumulants are the coefficients of Taylor expansion of the cumulant generating function [86, 87]. Consider the joint probability formed by the random vector $\mathbf{Z} = (Z_1, \ldots, Z_n)$, let the moment generating function be

$$\phi(\boldsymbol{\theta}) = E[e^{i\boldsymbol{\theta}^T \mathbf{Z}}]. \tag{2.26}$$

Then the cumulant generate functions is defined as

$$\psi(\boldsymbol{\theta}) = \ln \phi(\boldsymbol{\theta}), \tag{2.27}$$

where $\boldsymbol{\theta}$ is a vector denoted as $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$. Joint cumulants of multivariate random variables appear in the Taylor expansion of cumulant generate functions as

$$\psi(\boldsymbol{\theta}) = \sum_{r_1=0}^{\infty} \cdots \sum_{r_n=0}^{\infty} \kappa_{r_1,\ldots,r_n} \frac{(i\theta_1)^{r_1} \cdots (i\theta_n)^{r_n}}{r_1! \cdots r_n!}, \tag{2.28}$$

where $\kappa_{r_1,\ldots,r_n}$ is the cumulant with order r_i for each random variable Z_i.

The cumulants and moments can be exchanged to each other as [88]

$$K(Z_1, \ldots, Z_n) = \sum_{\sigma} (-1)^{|\sigma|-1} (|\sigma| - 1)! \prod_{a=1}^{|\sigma|} E\Big(\prod_{i \in \sigma_a} Z_i \Big), \tag{2.29}$$

$$E(Z_1, \ldots, Z_n) = \sum_{\sigma} \prod_{a=1}^{|\sigma|} \kappa_{\sigma_a}, \tag{2.30}$$

where $K(Z_1, \ldots, Z_n)$ is the cumulant of order n (noting that $Z_1, \ldots, Z_n$ can be redundant in this expression); σ is an arbitrary partition of set $\{1, \ldots, n\}$ and $|\sigma|$ is the number of blocks in partition σ. An alternative recursive form handy for computation is given as [89]

$$\mu_{r_1,\ldots,r_n} = \sum_{j_1}^{r_1} \cdots \sum_{j_{n-1}}^{r_{n-1}} \sum_{j_n}^{r_n - 1} \binom{r_1}{j_1} \cdots \binom{r_{n-1}}{j_{n-1}} \binom{r_n - 1}{j_n} \mu_{j_1,\ldots,j_n} \kappa_{r_1 - j_1, \ldots, r_n - j_n}, \tag{2.31}$$

$$\mu^*_{r_1,\ldots,r_n} = \sum_{j_1}^{r_1} \cdots \sum_{j_{n-1}}^{r_{n-1}} \sum_{j_n}^{r_n - 1} \binom{r_1}{j_1} \cdots \binom{r_{n-1}}{j_{n-1}} \binom{r_n - 1}{j_n} \mu^*_{j_1,\ldots,j_n} (-\kappa_{r_1 - j_1, \ldots, r_n - j_n}), \tag{2.32}$$

$$\kappa_{r_1,\ldots,r_n} = \sum_{j_1}^{r_1} \cdots \sum_{j_{n-1}}^{r_{n-1}} \sum_{j_n}^{r_n - 1} \binom{r_1}{j_1} \cdots \binom{r_{n-1}}{j_{n-1}} \binom{r_n - 1}{j_n} \mu^*_{j_1,\ldots,j_n} \mu_{r_1 - j_1, \ldots, r_n - j_n}, \tag{2.33}$$

where $\mu_{r_1,\ldots,r_n}$ and $\kappa_{r_1,\ldots,r_n}$ are the moments and cumulants, respectively, and $\mu^*_{r_1,\ldots,r_n}$ is defined as the moments corresponding to the negative cumulant generating function $-\psi(\boldsymbol{\theta})$ for the computation of recursive equation.

Consider the random field $\mathbf{Z}(\mathbf{u})$, spatial cumulants are defined on a family of random variables distributed to a spatial template $\boldsymbol{\tau} = \{\mathbf{h}_1, \ldots, \mathbf{h_n}\}$, and the explicit form can be derived from the above Equations from (2.29) to (2.33). For instance, the second-order cumulant is equivalent to the covariance and is given by

$$c_2^z(\mathbf{h}) = E[Z(\mathbf{u})Z(\mathbf{u}+\mathbf{h})] - E[Z(\mathbf{u})]^2. \tag{2.34}$$

Through several case studies with different geometry shapes of template, Dimitrakopoulos et al. [17] point out that spatial cumulants of different orders reveal various interrelations among multiple points, of which exhibit the so called duality relation between the geological patterns and spatial cumulants. In general, the mathematical entities of spatial cumulants provide an effective approach to characterize the complex spatial patterns, as a consequence they have been applied as an alternative tool to validate the realizations from stochastic simulations [11, 90]. A public domain software for geological pattern recognition using high-order spatial cumulants (HOSC) has been implemented by [91], and it is able to compute spatial cumulants either on regular or irregular grids. Li et al. [92] develop a GPU-based algorithm to calculate the spatial cumulants with considerable acceleration in computation, however the algorithm is not applicable for irregular grids.

2.4.2 HOSIM

Mustapha and Dimitrakopoulos [18] propose a high-order stochastic simulation (HOSIM) framework that aims to reproduce complex spatial patterns from a non-Gaussian random field. The sequential decomposition of CPDF is adopted in HOSIM similar to other sequential simulation methods. Major difference is that HOSIM does not assume a specific type of distribution. The derivation of CPDF is based on a Legendre polynomial series. Legendre polynomials can be defined by a differential equation as [93, 94]

$$P_m(z) = \frac{1}{2^m m!}\frac{d^m}{dz^m}[(z^2-1)^m] = \sum_i^m a_{i,m} z^i, \tag{2.35}$$

where $P_m(z)$ is the mth-degree Legendre polynomial and $z \in [-1, 1]$. The infinite sequence of polynomials form a complete orthogonal basis set on the domain $D = [-1, 1]$. The orthogonal property of the Legendre polynomials can be expressed as

$$\int_D P_m(z)P_n(z)dz = \begin{cases} 0 & m \neq n \\ \dfrac{2}{2m+1} & m = n \end{cases}, \tag{2.36}$$

and the norm of the Legendre polynomial $P_m(z)$ is

$$\| P_m \| = \sqrt{\frac{2}{2m+1}} \tag{2.37}$$

Mustapha and Dimitrakopoulos [18] develop their approximation of probability density function (PDF) by a truncated Legendre polynomial series with normalization as

$$f_w(z_0, z_1, \ldots, z_n) = \sum_{i_0=0}^{w} \cdots \sum_{i_{n-1}=0}^{i_{n-2}} \sum_{i_n=0}^{i_{n-1}} L_{\bar{i}_0,\bar{i}_1,\ldots,i_n} \overline{P}_{\bar{i}_0}(z_0) \cdots \overline{P}_{\bar{i}_{n-1}}(z_{n-1}) \overline{P}_{i_n}(z_n), \tag{2.38}$$

where $f_w(z_0, z_1, \ldots, z_n)$ is the approximation of PDF up to order w; $\overline{P}_{\bar{i}_0}(z_0), \ldots, \overline{P}_{i_n}(z_n)$ are the normalized Legendre polynomials and $\bar{i}_k = i_k - i_{k+1}$ for $k < n$. The coefficients $L_{\bar{i}_0,\bar{i}_1,\ldots,i_n}$, which is called Legendre cumulants in Mustapha and Dimitrakopoulos [18], can be derived from spatial cumulants as

$$L_{\bar{i}_0,\bar{i}_1,\ldots,i_n} = \int_D \overline{P}_{\bar{i}_0}(z_0) \cdots \overline{P}_{\bar{i}_{n-1}}(z_{n-1}) \overline{P}_{i_n}(z_n) f(z_0, z_1, \ldots, z_n) dz_0 \ldots dz_n = g(c_{\bar{i}_0,\bar{i}_1,\ldots,i_n}) \tag{2.39}$$

The local CPDF is obtained by Bayes' equation as

$$f_{Z_0}(z_0|\Lambda_0) = \frac{f_w(z_0, \Lambda_0)}{(\int_D f_w(z_0, z_1, \ldots, z_n) dz_0 \ldots dz_n)_{\Lambda_0}} \tag{2.40}$$

where $f_{Z_0}(z_0|\Lambda_0)$ is the CPDF and the denominator is the marginalized distribution over Z_0.

The rest main procedures of HOSIM follow as:

(1) The spatial cumulants are precomputed from the training image as well as the hard data and the results are stored in a tree data structure.

(2) For each node to be simulated, find the neighbor of conditioning data within the predefined template and estimate the local CPDF by Equations (2.38) and (2.40).

(3) Draw a random value from the local CPDF to the simulated node.

(4) Repeat from Step (2) until all nodes are simulated.

An important feature of HOSIM is that the spatial cumulants are computed from the sample data in priority and borrowed higher order statistics from the training image only when the estimation from the sample data is considered insufficient in case of scarcity of replicates. Thus, HOSIM is data-driven, differing from the conventional MPS methods that are training image driven and therefore hard to resolve the conflicts between the data and the training image. It has been shown in various publications that HOSIM is able to reproduce non-

Gaussian random fields with complex geological patterns and remains relatively insensitive to the inconsistency between TI and the sample data [18, 19]. However, the number of Legendre cumulants to the order w goes up to $\binom{N+w}{w}$ given a conditioning data of size N, which means that the computation increases fast as either N or w increasing. Another issue of HOSIM is that the positiveness is not ensured in the approximation of CPDF, although asymptotically convergence to a true distribution provided. Mustapha and Dimitrakopoulos [19] discard some high order terms to reduce the computational cost of Legendre cumulants. Although the revised HOSIM algorithm is feasible in computation, it also leads to a loss of accuracy of CPDF. Mustapha and Dimitrakopoulos [18] also mention the positiveness issue where they force correction on the negative parts of the CPDF to be non-negative with some post processing techniques. Mustapha and Dimitrakopoulos [95] propose another high-order simulation method approximating the CPDF with generalized Laguerre expansions where the coefficients are estimated from moments, however, the approach does not address the positiveness issue and interpolation by a quadratic polynomial is applied to correct the negative parts of CPDF. The positiveness problems are inevitable for all the approximation of CPDFs by typical polynomial expansions.

2.4.3 Simulation algorithms beyond HOSIM

Vargas-Guzmán [96] proposes a non-Gaussian simulation method for heavy tailed probability distributions with high-order cumulant parameters. Instead of a distribution-free framework adopted in HOSIM, Vargas-Guzmán [96] assumes that the non-Gaussian random variables follow a family of exponential power (EP) PDF and the random variables are decomposed into a set of random residual variables which can be written as power of the original random variables with high-order cumulants as the parameters. Several distributions including the distributions with skewed PDF are explicitly written as a EP distribution with high-order cumulants. Thus, it is straightforward to estimate parameters from the sample data and generate the prediction. Nevertheless, the derivation of the decomposition of PDF as residuals are provided only in one dimensional random variables and therefore not applicable to the 2D or 3D random fields. Abolhassani et al. [97] develop a new high-order simulation method assuming the CPDF from exponential family where the parameters are derived by maximum likelihood estimation. The CPDF is defined as function of a so called disparity vector which is in fact a new distance based on the high-order statistics between data event and replicates on TI. Eventually the replicates with more similar high-order statistics to the condition data are given more weights to formulate the likelihood function. As a consequence, the method is resistant to the conflicts between the hard data and the TI. However, a weak point this method is that it still relies on replicates from the TI to obtain the parameter estimation

except the impacts of replicates with lower similarity in high-order statistics to the data event are filtered out, thus the stability of the estimation may demand larger TI or more than one TI. Minniakhmetov and Dimitrakopoulos [98] propose a new high-order simulation framework which is able to generate realizations for categorical random variables. In their method, the high order spatial indicator moments are approximated by B-spline functions with the experimental statistics calculated from the replicates in the hard data, and the normalized B-spline functions are used as the approximation of CPDF to draw random values for the simulation. The method is data driven without referring to a TI, whereas it is difficult to extend to simulation of continuous attributes.

2.4.4 Extension to multiple variables

The extension of high-order simulation methods to multiple variables are relatively few. A most recent development of joint high-order simulation with decorrelation of high-order spatial statistics is proposed by Minniakhmetov and Dimitrakopoulos [99]. In this method, Minniakhmetov and Dimitrakopoulos [99] develop a decorrelation technique with diagonal dominant cumulants aiming to approach the statistical independence of the decomposed factors. A linear transformation is assumed to obtain the new factors $\mathbf{Y}(\mathbf{u})$ from the random field $\mathbf{Z}(\mathbf{u})$ as

$$\mathbf{Y}(\mathbf{u}) = A\mathbf{Z}(\mathbf{u}). \tag{2.41}$$

The diagonal dominant factorization can be expressed as a minimization problem with the objective function as

$$\min \sum_{d} \alpha_d F_d(A), \tag{2.42}$$

and $F_d(A)$ is defined as

$$F_d(A) = \sum_{k_0} 1 + \frac{\sum_{\text{non-diagonal}} \parallel Cum(\mathbf{Y}_{k_0}(\mathbf{u}), \mathbf{Y}_{k_1}(\mathbf{u}), \ldots, \mathbf{Y}_{k_{d-1}}(\mathbf{u})) \parallel_2}{\parallel Cum(\mathbf{Y}_{k_0}(\mathbf{u}), \mathbf{Y}_{k_0}(\mathbf{u}), \ldots, \mathbf{Y}_{k_0}(\mathbf{u})) \parallel_2} \tag{2.43}$$

The main idea of the algorithm is to find a linear transformation such that the diagonal elements dominate the cumulants tensor. HOSIM is used to generate realizations for each decomposed factor and the results are back transformed to the original data space to get the joint simulations of multiple variables. Currently, the algorithm only considers decorrelation at lag zero. To consider impact of cross-cumulants at various lags, the objective function needs to be revised to include more terms which may increase the complexity of solving the minimization problem.

2.5 Learning-Based Stochastic Simulation and Kernels

The development of artificial intelligence over the past decades leads to various machine learning methods and their vast applications in engineering domains. State-of-the-art machine learning techniques have been integrated to uncertainty quantification to develop learning-based stochastic simulation methods. As an early attempt to apply machine learning method to stochastic simulation, Caers [100] proposes a simulation method using neural network to predict the conditional probability distribution with incorporation of multiple point statistics. The Metropolis-Hasting sampler [101] is used to generate the random values to be simulated through a iterative updating procedure. The target probability density function is expressed as a convex combination of certain type of density functions and the related parameters are learned from the replicates with a given spatial template retrieved from the training image based on the Expectation-Maximization (EM) algorithm [102]. More recently, a deep learning technique called generative adversarial networks (GAN) [103] has been substantially studied and applied to stochastic simulation [104–107]. The architecture of GAN contains two main components, a generator G and a discriminator D each corresponded to a neural network. Given that a set of sample images which are labeled as the real data, the images created by the generator G are labeled as the fake data. A latent space $\mathbf{Z}$ consists of a set of independent random variables, from which the random samples $\mathbf{z}$ are drawn as the input for the generator G. The discriminator D is trained to distinguish the two categories of images and label them correctly as much as possible. On the other hand, the generator G is trained to create images as close to the sample images as possible so that discriminator D can be fooled by the fake data. This adversary training is achieved by solving a minimization-maximization problem as

$$\min_G \max_D E_{\mathbf{x}\sim p_{data}(\mathbf{x})}\Big[\log D(\mathbf{x})\Big] + E_{\mathbf{z}\sim p(\mathbf{z})}\Big[\log\Big(1 - D\Big[G(\mathbf{z})\Big]\Big)\Big], \tag{2.44}$$

where $p_{data}(\mathbf{x})$ corresponds to the probability distribution of the real data, and $p(\mathbf{z})$ corresponds the probability distribution of the random variables in the latent space. In the context of multiple point simulation, the training images act as the real data and the generated realizations are created by the generator G. While the GAN converges after training, the generated realizations should get close to the underlying distribution of the training images. As many of the stochastic simulation methods using GANs are unconstrained by the sample data, Dupont et al. [105] propose a simulation method conditioned to the physical measurements with GAN. They introduce two loss functions, namely prior loss and the context loss, to balance the reproduction of the statistics from the training images and the sample data. The prior loss penalizes the deviation from the distribution of the training images and the context loss penalized the mismatch between the generated data and the actual measurement

at the locations of the sample data. Similar loss functions are also introduced in [107] to develop stochastic subsurface model reconstruction using GAN. A recent research in Avalos and Ortiz [108] also uses the convolutional neural network (CNN) [109] to develop multiple point simulation method, where the spatial patterns are captured by the features extracted by the convolutional layers for learning the probability distribution of spatial attributes given a certain spatial template.

The kernel methods provide another way to represent the original data as features in a high-dimensional feature space, and they are widely used in statistical learning [110, 111]. Some earliest application of kernel to probability density estimation can date back to Parzen [112] and the related methods are called kernel density estimation (KDE) methods[113–115]. In general, a kernel function is positive definite and has the so-called reproducing property [116], from which a so-called reproducing kernel Hilbert space (RKHS) is determined. In machine learning field, the so-termed feature space resembles the kernel Hilbert space by taking the elements from the original data space into the kernel space as features. The features usually carries higher dimensional information than the raw data, however, the similarity between the features can be expressed by the kernel functions. In terms of stochastic simulation, the covariance function is a kernel and defines a dual kernel space of the original data space [117]. In the multiple point simulation methods, kernels have been used to measure the similarity between spatial patterns by feature mapping [67, 118], nevertheless, these applications are limited in the sense that learning a random field model is not under consideration.

CHAPTER 3 GENERAL ORGANIZATION OF THE DOCUMENT

This thesis research develops new stochastic simulation methods to quantify spatial uncertainty incorporating the high-order spatial statistics of the available data, aiming for better performance in reproducing the underlying complex patterns in various natural phenomena. The literature review reveals the evolution of stochastic simulation methods from the conventional second-order stochastic simulation methods to the multiple-point simulation methods, as well as more recent high-order simulation methods. In general, the trending of the stochastic simulation methods shifts from the Gaussianity to non-Gaussianity, and from pair-wise correlations to multiple-point spatial continuity. The new paradigm of high-order stochastic simulation provides a distribution-free random field model accounting for high-order spatial statistics, distinguished from other models. How to effectively and efficiently incorporate high-order spatial statistics, however, poses challenges from both the theoretical and the computational aspects, as well as the practical challenges in real-life applications.

To reach the research goal, the thesis contains four different chapters as the main content to deliver the major research objectives including the theoretical development, computational model, and the practical aspects. Chapter 4 proposes a new computational model of high-order simulation based on spatial Legendre moments. Although the proposed computational model is derived from the concept of spatial Legendre moments, the explicit computation of the empirical high-order spatial statistics is avoided in the related numerical equation. Instead, the approximation of probability density function is written in a kernel-like form with spatial statistics of different orders incorporated in a unified function. The proposed computational model not only improves the computational efficiency of utilizing high-order spatial statistics of the available data during the simulation, but also leads to a concept of a new kernel function, the so-called spatial Legendre moment kernel (SLM-kernel) proposed in Chapter 5.

The proposed SLM-kernel is proven to be positive definite in Chapter 5 and, thus possesses the so-called reproducing property to construct a reproducing kernel Hilbert space (RKHS). A feature mapping is also defined to map the replicates of the data events into the new kernel space. The high-order spatial statistics of the available data are encapsulated in the empirical kernel statistics. In the same manner, the probability distributions of the related random field model are embedded into the same kernel space in the form of expected kernel statistics. A new statistical learning framework is proposed for high-order stochastic simulation through a kernelized learning algorithm.

In Chapter 6, the concept of aggregated kernel statistics is proposed to utilize the high-order spatial statistics from the ensemble set of the replicates retrieved from the sample data with different spatial configurations. The statistical learning framework proposed in Chapter 6 is further extended to incorporate the aggregated kernel statistics. This extension allows sparse data learning from the relatively sparse sample data and, thus leads to a new training-image free, high-order simulation method.

In Chapter 7, the statistical learning framework is adopted to accommodate high-order spatial information at multiple scales. Specifically, a learning algorithm is proposed to incorporate the high-order spatial statistics at coarse scales from the sample data, while complement the high-order spatial statistics at finer scales with the informatin from the training image.

A general discussion on top of the main developments in this thesis is presented in Chapter 8. Chapter 9 concludes the major contributions of the thesis research. Limitations regarding the thesis research is discussed and potential future research is presented.

CHAPTER 4 ARTICLE 1: A NEW COMPUTATIONAL MODEL OF HIGH-ORDER STOCHASTIC SIMULATION BASED ON SPATIAL LEGENDRE MOMENTS

Abstract: Multiple-point simulations have been introduced over the past decade to overcome the limitations of second-order stochastic simulations in dealing with geologic complexity, curvilinear patterns and non-Gaussianity. However, a limitation is that they sometimes fail to generate results that comply with the statistics of the available data while maintaining the consistency of high-order spatial statistics. As an alternative, high-order stochastic simulations based on spatial cumulants or spatial moments have been proposed; however, they are also computationally demanding, which limits their applicability. The present work derives a new computational model to numerically approximate the conditional probability density function (*cpdf*) as a multivariate Legendre polynomial series based on the concept of spatial Legendre moments. The advantage of this method is that no explicit computations of moments (or cumulants) are needed in the model. The approximation of the *cpdf* is simplified to the computation of a unified empirical function. Moreover, the new computational model computes the *cpdf*s within a local neighborhood without storing the high-order spatial statistics through a predefined template. With this computational model, the algorithm for the estimation of the *cpdf* is developed in such a way that the conditional cumulative distribution function (*ccdf*) can be computed conveniently through another recursive algorithm. In addition to the significant reduction of computational cost, the new algorithm maintains higher numerical precision compared to the original version of the high-order simulation. A new method is also proposed to deal with the replicates in the simulation algorithm, reducing the impacts of conflicting statistics between the sample data and the training image. A brief description of implementation is provided, and for comparison and verification, a set of case studies are conducted and compared with the results of the well-established multi-point simulation algorithm, *filtersim*. This comparison demonstrates that the proposed high-order simulation algorithm can generate spatially complex geological patterns while also reproducing the high-order spatial statistics from the sample data.

Keywords: High-order stochastic simulation, multi-point statistics, spatial moments, Legendre polynomials

---

Published: Yao L, Dimitrakopoulos R, Gamache M (2018) A new computational model of high-order stochastic simulation based on spatial Legendre moments. Math Geosci 50 (8):929-960

4.1 Introduction

For the past several decades, stochastic simulations have been used to quantify spatial uncertainty in earth science applications. Traditionally, stochastic models are built on the basis of the Gaussian distribution and two-point statistics, where covariance or variograms are used to capture the spatial correlations [22, 23, 26, 40]. The limitations of the existing two-point simulation methods have been reported in various publications [11–14, 26, 84], which are mostly related to the poor reproduction of spatial distributions while dealing with the complex spatial patterns, spatial connectivity of extreme values and non-Gaussianity. To reflect the complex geological patterns, multi-point statistics have to be introduced instead of conventional two-point statistics. Guardiano and Srivastava [12] propose a multiple-point simulation (*mps*) framework and the concept of the training image (TI). The primary difference between *mps* and two-point simulations is that the conditional cumulative distribution functions (*ccdf*) are built on empirical estimations of conditional probabilities with multiple-point configurations, which is equivalent to solving a normal equation according to the Bayes' rule. Strebelle [57] formalizes the method and develops the first computationally-efficient implementation. For over a decade, research has been focused on various issues around *mps* algorithms, such as computational efficiency and various patch-based extensions [60, 62, 63, 66, 67, 70, 75, 119–124]. In general, these *mps* methods are TI-based, and their statistics are estimated from distributions of replicates of data events in the training image. Their main drawbacks are: (1) the high-order statistics are partially and indirectly considered; (2) the methods are not driven by a consistent mathematical framework; and, (3) since they are TI-driven, they may not generate results that comply with the statistics of actual available data. The latter shortcoming becomes distinctly clear in mining applications, where dense data sets are used [125, 126].

As an alternative, a high-order simulation framework with mathematical consistency is proposed with the introduction of a new concept of spatial cumulants [17]. The so-called high-order simulation algorithm (*hosim*) and its implementation are developed by [19, 91]. In this algorithm, the conditional probability density function (*cpdf*) is approximated by a multivariate expansion with coefficients expressed in terms of spatial cumulants. The *hosim* algorithm is extended mostly recently to deal with the joint simulation of multiple variables as well as the simulation of categorical data [98, 127]; other extensions are approximating the *cpdf* with different types of orthogonal polynomial bases, such as expansion series with Laguerre polynomials and Legendre-Like spline polynomials [95, 128]. However, the related calculations are computationally demanding, since the number of spatial cumulants involved in the series increases exponentially either as the order of cumulants or the quantity of con-

ditioning data increases. In [19], some terms of the expansion series have to be discarded to obtain computational feasibility, which compromises the accuracy of the approximated *cpdf*. In addition, the computational cost limits the approach for larger-scale applications.

To take full advantage of the high-order simulation, that is, its data-driven aspect and no presumption of data distribution, and address the computational difficulties, a new stochastic simulation algorithm based on high-order spatial Legendre moments is presented herein. Rather than just a mathematical equivalency of the previous model of the high-order simulation, the approximation of *cpdf* by Legendre polynomial series is reformulated under the framework of the sequential simulation, leading to a much more concise form of the computational model. In this new method, all explicit calculations of moments are encapsulated in a unified function to derive the *cpdf*, cutting down the previous complex computations into a few iterations of simple operations with polynomial time. Moreover, there is no predefined template configuration in the new algorithm, as required for the normal *mps* methods and the previous *hosim* model. The spatial configuration of the template will instead depend on the local neighborhood of the node to be simulated; note that there is no need to store the intermediate results in a tree as in most of the *mps* methods including the previous *hosim*. The variable template also has the advantage of simultaneously capturing the spatial patterns either in local scale or global scale.

The remainder of the paper continues with Section 4.2, which describes the stochastic model based on the concepts of high-order spatial Legendre moments. Section 4.3 develops the computational model as a statistical function. Section 4.4 describes the new proposed high-order simulation algorithm and analyzes the computational complexity. Section 4.5 explores the implementation of the new high-order simulation algorithm. Section 4.6 shows the examples to assess the new method and compare it with *filtersim*. Finally, conclusions and future research are presented in Section 4.7.

4.2 Stochastic Model of High-Order Simulation with Spatial Legendre Moments

4.2.1 Sequential simulation

In this paper, the stochastic model is discussed specifically under the sequential simulation framework [23–25]. Sequential simulation aims to reproduce spatial properties sequentially by decomposing the multivariate conditional distributions into a set of univariate distributions. Considering a stationary and ergodic random field $\boldsymbol{Z}(\boldsymbol{u})$, let $Z(\boldsymbol{u}_1), \ldots, Z(\boldsymbol{u}_N)$ be a set of random variables with locations at $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_N$, respectively. Then, the N random variables $Z(\boldsymbol{u}_1), \ldots, Z(\boldsymbol{u}_N)$ constitute a joint multivariate distribution. In terms of the stochastic

simulation, it is supposed that realizations are to be generated from $Z(\boldsymbol{u}_1), \ldots, Z(\boldsymbol{u}_N)$, and the available data set are $\Lambda_0 = \{\zeta(\boldsymbol{u}_1'), \ldots, \zeta(\boldsymbol{u}_n')\}$, where $\zeta(\boldsymbol{u}_i')$ is the sample data at the location $\boldsymbol{u}_i'$ for $i = 1, \ldots, n$, and n is the number of sample data in total. For simplification, $Z(\boldsymbol{u}_1), \ldots, Z(\boldsymbol{u}_N)$ are alternatively written as $Z_1, \ldots, Z_N$, and a similar simplification of notation applies in the context of a random field. Following the above notation, the stochastic simulation of the random field is based on the sampling from the N-variate probability distribution posterior to the data set Λ_0, which can be characterized by a conditional cumulative distribution function (*ccdf*) as $F_{\boldsymbol{Z}}(z_1, \ldots, z_N|\Lambda_0)$ or by a probability density function (*cpdf*) as $f_{\boldsymbol{Z}}(z_1, \ldots, z_N|\Lambda_0)$. The joint *cpdf* $f_{\boldsymbol{Z}}(z_1, \ldots, z_N|\Lambda_0)$ can be decomposed into the product of a series of univariate *cpdf*s (Rosenblatt 1952; Johnson 1987) as

$$f_{\boldsymbol{Z}}(z_1, \ldots, z_N|\Lambda_0) = f_{Z_1}(z_1|\Lambda_0) \cdots f_{Z_N}(z_N \mid \Lambda_{N-1}), \tag{4.1}$$

where $\Lambda_i (i = 1, \ldots, N-1)$ are a series of sets and $\Lambda_i = \Lambda_{i-1} \cup \{\zeta(\boldsymbol{u}_i)\}$, $i = 1, \ldots, N$, where $\zeta(\boldsymbol{u}_i)$ is the value drawn from the conditional probability distribution with a density function described as $f_{Z_i}(z_i|\Lambda_0)$.

The basic idea of sequential simulation is to sequentially draw random values from the decomposed univariate *cpdf*s through a random path that visits all the nodes to be simulated. Irrespective of the node's location corresponding to the sequence number, there is no difference in the sampling procedures. Without loss of generality, the *cpdf* in every single sampling procedure can be symbolized uniformly as $f_{Z_0}(z_0|\Lambda)$, where Z_0 means the current simulating node and Λ means the set of conditioning data around Z_0's location $\boldsymbol{u}_0$. Considering the computational intensity and the statistical relevancy, the conditioning data are usually confined to a neighborhood closest to the simulation node instead of taking account of all available data on the whole domain of the random field. For more details on this screen-effect approximation, the reader is referred to Dimitrakopoulos and Luo [28].

An algorithmic description of sequential simulation can be summarized as the following steps:

1. Draw a random path to visit all the N nodes to be simulated.
2. Starting from $i = 1$ and for each node $Z(\boldsymbol{u}_i)$, derive the conditional probability cumulative distribution $F_{Z_i}(z_i|\Lambda_{i-1})$ or the density function $f_{Z_i}(z_i|\Lambda_{i-1})$.
3. Draw a random value $\zeta(\boldsymbol{u}_i)$ from the conditional probability distribution in Step 2 and update the conditioning data by adding the node value $\zeta(\boldsymbol{u}_i)$ in to the current data set Λ_i.
4. Repeat from Step 2 until all the nodes are visited.

4.2.2 High-order spatial Legendre moments

In probability theory, moments are defined as expectations of integer power functions of a random variable. Given a random variable Z in probability space (Ω, F, P), suppose that the density of probability measure P is a continuous function $f_Z(z)$. The moment of order w is defined as

$$\mathrm{Mom}_Z(w) = E[Z^w] = \int_\Omega [Z(\omega)]^w dP(\omega) = \int_{\mathbb{R}} z^w f_Z(z)\, dz. \tag{4.2}$$

The moments of random vector $\boldsymbol{Z} = [Z_{\mathbf{0}}, \ldots, Z_N]$ with a multivariate density $f_{\boldsymbol{Z}}(z_0, \ldots, z_N)$ defined similarly as

$$\mathrm{Mom}_{\boldsymbol{Z}}(w_0, \cdots, w_N) = E[z_0^{w_0} \cdots z_N^{w_N}] = \int_{\mathbb{R}^N} z_0^{w_0} \cdots z_N^{w_N} f_{\boldsymbol{Z}}(z_0, \ldots, z_N)\, dz_0 \cdots dz_N, \tag{4.3}$$

where $w_i (i = 0, \cdots, N)$ are the orders of moments for the i-th element of vector $\mathbf{Z}$. The spatial moments of a discrete random field $\mathbf{Z}=[Z(\boldsymbol{u}_0), \ldots, Z(\boldsymbol{u}_N)]$ are functions of spatial location variables $\boldsymbol{u}_0, \ldots, \boldsymbol{u}_N$. Assuming the random field $\mathbf{Z}(\boldsymbol{u})$ is stationary and ergodic, the spatial moments of $\mathbf{Z}(\boldsymbol{u})$ can be expressed as functions of distance vectors, and thus they are independent of the locations. These distance vectors, which keep the spatial configuration of a center node and nodes within its neighborhood, can be expressed using a spatial template $\mathbf{T}$ (Figure 4.1). The terminologies of the spatial template $\mathbf{T}$ and data events [18, 57] are the following:

(i) Spatial template $\mathbf{T}$: geometry defined by N distance vectors $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N)$ from the center node $\boldsymbol{u}_0$, $\mathbf{T} = \{\boldsymbol{u}_0, \boldsymbol{u}_0 + \boldsymbol{h}_1, \ldots, \boldsymbol{u}_0 + \boldsymbol{h}_N\}$.

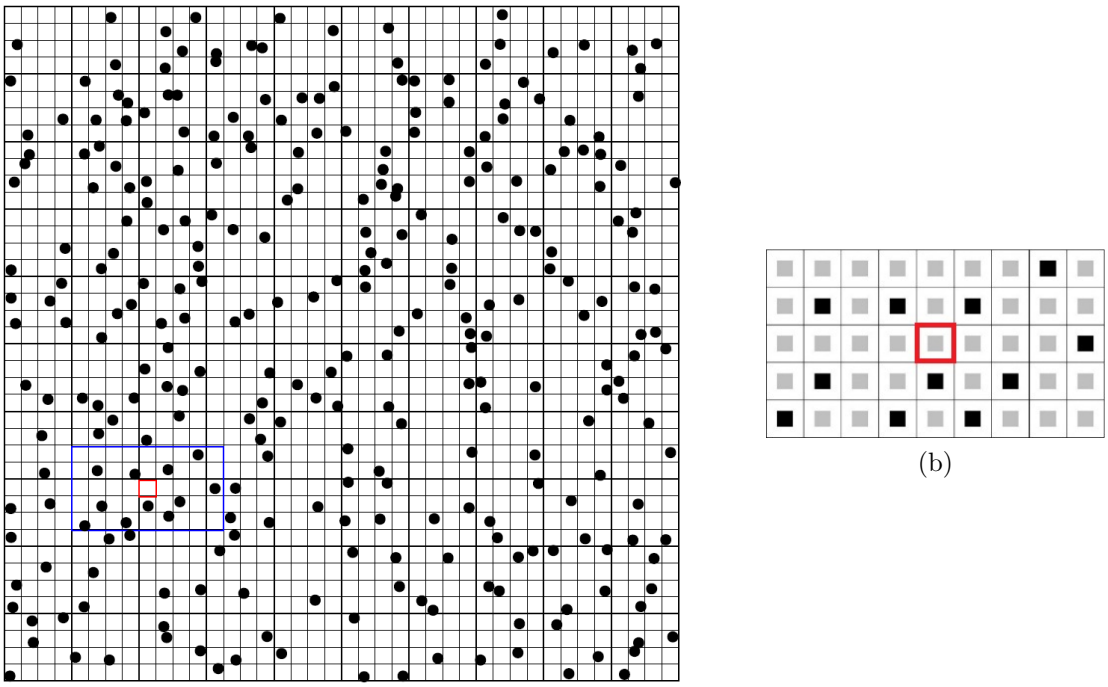
(ii) Data events: outcomes of the random field in the spatial template $\mathbf{T}$. Specifically, the data events are conditioning data set Λ in the present work.

The spatial moments of a random field $\mathbf{Z}$ in a template $\mathbf{T}$ can be expressed element-wise as

$$\mathrm{Mom}_{\mathbf{Z}}^{\mathbf{T}}(w_0, \cdots, w_N) = \mathrm{E}[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N; Z_0^{w_0} \cdots Z_N^{w_N}], \tag{4.4}$$

where $\mathrm{Mom}_{\mathbf{Z}}^{\mathbf{T}}$ is the moment function of $\mathbf{Z}$ in the spatial template $\mathbf{T}$, $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N)$ are the distance vectors to represent the geometry of $\mathbf{T}$ and w_i are the orders of the moments with each random variable $Z(\boldsymbol{u}_i)$ $(i = 1, \ldots, N)$.

The Legendre polynomials are used here to further define the concept of spatial Legendre moments. Legendre polynomials are one kind of special math functions defined on the interval



(a) (b)

Figure 4.1 (a) A size 40×40 grid to be simulated with a 9×5 template overlaid on the current visiting node; (b) Spatial template **T** and a certain data event in **T**, the center square is the node to be simulated; the black squares are the conditioning data

$[-1, 1]$, which can be expressed using Rodrigues' formula [94]

$$P_m(z) = \frac{1}{2^m m!} \frac{d^m}{dz^m} \left[\left(z^2 - 1 \right)^m \right], \tag{4.5}$$

where$P_m(z)$ is the mth-degree Legendre polynomial. The infinite sequence of polynomials forms a complete orthogonal basis set on the domain $D = [-1, 1]$. The orthogonal property of the Legendre polynomials can be expressed as

$$\int_D P_m(z) P_n(z) dz = \begin{cases} 0 & m \neq n \\ \frac{2}{2m+1} & m = n \end{cases} \tag{4.6}$$

and the norm of the Legendre polynomial $P_m(z)$ is

$$\| P_m \| = \sqrt{\frac{2}{2m+1}}. \tag{4.7}$$

With a simple substitution of polynomials in moment function (4.4) into Legendre polyno-

mials, the spatial Legendre moments are defined as

$$L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} = \prod_{i=0}^{N} \left(w_i + \frac{1}{2}\right) \cdot E\left[h_1, \ldots, h_N; P_{w_0}\left(z_0\right) P_{w_1}\left(z_1\right) \cdots P_{w_N}\left(z_N\right)\right], \tag{4.8}$$

where $L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N}$ are Legendre moments defined on the spatial template $\mathbf{T}$; the extra coefficient $\left(w_i + \frac{1}{2}\right)$ on the right-hand side of the equation is intentionally introduced as a normalization term for convenience of the later computation (see Appendix for the details).

4.2.3 Multivariate expansion series of joint probability density function

A piecewise continuous function $f(z)$ defined on the interval $[-1, 1]$ can be written as a series of Legendre polynomials

$$f\left(z\right) = \sum_{m=0}^{\infty} L_m P_m\left(z\right). \tag{4.9}$$

Likewise, the expansion of a multivariate function $f(z_0, z_1, \ldots, z_N)$ can defined on an $N+1$-dimensional domain in the same way. Specifically, suppose that the multivariate function is a density function related to the joint distribution of random variables on a spatial template $\mathbf{T}$. The density function can be expanded into Legendre polynomial series in terms of Legendre spatial moments and Legendre polynomials as (see Appendix for the details)

$$f\left(z_0, z_1, \ldots, z_N\right) = \sum_{w_0=0}^{\infty} \sum_{w_1=0}^{\infty} \cdots \sum_{w_N=0}^{\infty} L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} P_{w_0}\left(z_0\right) P_{w_1}\left(z_1\right) \cdots P_{w_N}\left(z_N\right). \tag{4.10}$$

In practice, the above infinite series, Equation (4.10), are truncated at a certain order W, thus leading to the approximated density function

$$f\left(z_0, z_1, \ldots, z_N\right) \approx f_W\left(z_0, z_1, \ldots, z_N\right) = \sum_{w_0=0}^{W} \sum_{w_1=0}^{W} \cdots \sum_{w_N=0}^{W} L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} \prod_{i=0}^{N} P_{w_i}\left(z_i\right). \tag{4.11}$$

From definition (4.8), the spatial Legendre moments can be explicitly derived as

$$L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} = \int_D \prod_{i=0}^{N} \left[\left(w_i + \frac{1}{2}\right) P_{w_i}\left(z_i\right)\right] f\left(z_0, z_1, \ldots, \ z_N\right) dz_0 dz_1 \cdots dz_N. \tag{4.12}$$

Experimentally, if there are M replicates of data events associated with template $\mathbf{T}$ found in

the training image, the spatial Legendre moments can be calculated as

$$\tilde{L}^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} = \frac{1}{M} \sum_{t=1}^{M} \prod_{i=0}^{N} \left(w_i + \frac{1}{2} \right) P_{w_i} (\zeta_{t,i}), \tag{4.13}$$

where $\zeta_{t,i}$ are the data values of replicates in the template $\mathbf{T}$, t is the sequence number of replicates and i is sequence number of random variables.

4.3 Computational Model

Combining the Equations (4.10) to (4.13), the empirical joint *pdf* can be derived as

$$\begin{aligned} \tilde{f}(z_0, z_1, \ldots, z_N) \approx \tilde{f}_W(z_0, z_1, \ldots, z_N) &= \frac{1}{M} \sum_{t=1}^{M} \sum_{w_0=0}^{W} \sum_{w_1=0}^{W} \cdots \prod_{i=0}^{N} \left[\left(w_i + \frac{1}{2} \right) P_{w_i}(\zeta_{t,i}) P_{w_i}(z_i) \right] \\ &= \frac{1}{M} \sum_{t=1}^{M} \prod_{i=0}^{N} \left[\sum_{w=0}^{W} \left(w + \frac{1}{2} \right) P_w(\zeta_{t,i}) P_w(z_i) \right] \end{aligned} . \tag{4.14}$$

Equation (4.14) gives a unified computational model of empirical estimation of the density function on the spatial template $\mathbf{T}$, noticing that, in the right-hand side of the equation, the subscript i of w_i is dropped because of the symmetry of computation.

Now let's consider the *cpdf* $f_{Z_0}(z_0 \mid \Lambda)$ of a single sampling step in sequential simulation (ref. Section 4.2.1). The joint *pdf* can be marginalized from the Equation (4.14) to get the marginal *pdf* of conditioning random variables. To specify the difference between the empirical models and theoretical models in Equations (4.10)–(4.11), $\tilde{f}$ and $\tilde{f}_W$ specifically denote the experimental function corresponding to probability density function f and its Legendre polynomial series truncated at order W, respectively.

For convenience, denote functions $X_t(z_i)$ as

$$X_t(z_i) = \sum_{w=0}^{W} \left(w + \frac{1}{2} \right) P_w(\zeta_{t,i}) P_w(z_i). \tag{4.15}$$

Then, Equation (14) can be rewritten as

$$\tilde{f}_W(z_0, z_1, \ldots, z_N) = \frac{1}{M} \sum_{t=1}^{M} X_t(z_0) \prod_{i=1}^{N} X_t(z_i). \tag{4.16}$$

The result of integration of $X_t(z)$ over $[-1, 1]$ can be derived from the orthogonal properties

of Legendre polynomials as

$$\int_{-1}^{1} X_t\left(z_i\right) dz_i = 1. \tag{4.17}$$

In fact, Equations (4.16) and (4.17) ensures that the integral of the approximated probability density function to be 1, a necessary property of probability density.

Followed by the marginalization and Equation (4.17), the empirical density of marginal distribution on the random variables $z_1, \ldots, z_N$ is

$$\tilde{f}_W\left(z_1, \ldots, \ z_N\right) = \frac{1}{M} \sum_{t=1}^{M} \prod_{i=1}^{N} X_t\left(z_i\right). \tag{4.18}$$

From Equations (4.16) and (4.18) and considering the relation between the conditional probability density function and the joint probability density function, one can derive

$$f\left(z_0 \mid \Lambda\right) \approx \tilde{f}_W\left(z_0 \mid \Lambda\right) = \frac{\sum_{t=1}^{M} X_t\left(z_0\right) \cdot \prod_{i=1}^{N} X_t\left(\zeta_i\right)}{\sum_{t=1}^{M} \prod_{i=1}^{N} X_t\left(\zeta_i\right)}, \tag{4.19}$$

which provides a concise computational model of the *cpdf*.

The above development provides a theoretical equivalency of the approximation of *cpdf* by truncated Legendre series, which was proposed in Mustapha and Dimitrakopoulos [18, 19]. However, the new reformulated model in the current paper leads to a different stochastic simulation method in view of the related computational aspects. The advantage of the new model represented by Equation (4.19) is that no explicit computations of moments or cumulants are needed. In addition, the new model is computationally more accurate than the *hosim* program in Mustapha and Dimitrakopoulos [19], in which some terms have to be dropped from the full expansion of Legendre series in the form of spatial cumulants to gain computational efficiency.

4.4 Algorithm Description and Computational Analysis

4.4.1 Algorithm for computing cpdf

From the Equations (4.17) to (4.19), it can be easily shown that

$$\int_{-1}^{1} \tilde{f}_W\left(z_0 \mid \Lambda\right) dz_0 = 1. \tag{4.20}$$

As $X_t\left(\zeta_{t,i}\right)$ is a constant from the Equation (4.15), and from Equations (4.15) and (4.19), it is obvious that $\tilde{f}_W\left(z_0 \mid \Lambda\right)$ can be expressed as the summation of a series of Legendre

polynomials, that is

$$\tilde{f}_W(z_0 \mid \Lambda) = \sum_{w=0}^{W} c_w P_w(z_0), \tag{4.21}$$

where $c_w (w = 1, \ldots, W)$ are constants which can be conveniently computed as shown in the following Algorithm 4.1.

By the property of Legendre polynomial that $P_0(z) = 1, \forall z \in [-1, 1]$, combined with Equations (4.15) and (4.21), the computation of coefficients $c_w (w = 1, \ldots, W)$ can be divided into the computation of functions $X_t(z_i)$ over the nodes of each replicate. Especially, the first term of c_w is always fixed as $c_0 = \frac{1}{2}$.

Algorithm 4.1 Calculation of a *cpdf*

Data:
(1) data event (conditioning data): $\Lambda = \{\zeta_1, \ldots, \zeta_N\}$;
(2) replicates of data events: $\zeta_{t,i},\ t = 1, \ldots, M; i = 1, \ldots, N$;
(3) maximum order: W
Result: conditional probability density function
// Initialize the coefficients
for $w = 0$ **to** W **do**
  $c[w] = 0$;
end for
for $t = 1$ **to** M **do**
  //Computation of the function $X_t(z_0)$
  for $w = 0$ **to** W **do**
    $X[w] = \left(w + \frac{1}{2}\right) \cdot P_w(\zeta_{t,0})$;
  end for
  //According to Equation (4.19), the product $\prod_{i=1}^{N} X_t(\zeta_i)$ needs to be computed.
  //The product is initialized as 1 before the calculation of $X_t(\zeta_i)$
  $X\_Prod = 1$;
  // Update the product $\prod_{i=1}^{N} X_t(\zeta_i)$ by computing the function $X_t(\zeta_i)$
  for $i = 1$ **to** N **do**
    $X_t = 0$;
    for $w = 0$ **to** W **do**
      $X_t = X_t + \left(w + \frac{1}{2}\right) \cdot P_w(\zeta_{t,i}) \cdot P_w(\zeta_i)$
    end for
    $X\_Prod = X\_Prod \cdot X_t$;
  end for
  for $w = 0$ **to** W **do**
    $X[w] = X[w] \cdot X\_Prod$;
    $c[w] \ += X[w]$;
  end for
end for
//Note that the denominator in Equation (4.19) $\sum_{t=1}^{M} \prod_{i=1}^{N} X_t(\zeta_i)$ is actually equal to $2 \cdot c[0]$
$denom = 2 \cdot c[0]$;
for $w = 0$ **to** W **do**
  $c[w] = c[w] / denom$;
end for
END OF ALGORITHM

4.4.2 Recursive algorithm for computing a ccdf

From the results of Algorithm 4.1, the conditional probability density function can be expressed as

$$f\left(z_0 \mid \Lambda\right) = \frac{1}{2} + \sum_{w=1}^{W} c_w P_w\left(z_0\right). \tag{4.22}$$

The coefficient $c_0 = \frac{1}{2}$ is taken out from the summation in Equation (4.22) so that the Bonnet's recursion relation of Legendre polynomials can be smoothly applied in the followed derivation.

According to the Bonnet's recursion relation of Legendre polynomials

$$\left(2w+1\right) P_w\left(z\right) = \frac{d}{dz}\left[P_{w+1}\left(z\right) - P_{w-1}\left(z\right)\right], \tag{4.23}$$

the following equation can be derived

$$\left(2w+1\right) \int_{-1}^{z_0} P_w\left(z\right) dz = P_{w+1}\left(z_0\right) - P_{w-1}\left(z_0\right). \tag{4.24}$$

Therefore, the conditional cumulative distribution function (*ccdf*), $F\left(z_0 \mid \Lambda\right)$, can be deduced as

$$\begin{aligned} F\left(z_0 \mid \Lambda\right) =& \int_{-1}^{z_0} f\left(z_0 \mid \Lambda\right) dz \\ =& \frac{1}{2} + \frac{1}{2} z_0 + \sum_{w=1}^{W} \frac{c_w}{2w+1}\left[P_{w+1}\left(z_0\right) - P_{w-1}\left(z_0\right)\right]. \\ =& \sum_{w=0}^{W+1} d_w P_w\left(z_0\right) \end{aligned} \tag{4.25}$$

As can be seen from Equation (4.25), the *ccdf* is also expressed as the summation of the univariate Legendre polynomials, with the order of the Legendre polynomials increasing by one because of the integration. Furthermore, the new coefficients $d_w (w = 0, \ldots, W, W+1)$ now can be computed through Equation (4.25) in an iterative way, as shown in Algorithm 4.2.

4.4.3 Computational complexity

The most computationally demanding part of the high-order simulation algorithm is to calculate the Legendre series coefficients, which is the basis for estimating the conditional probability density functions. Considering that the conditional probability density functions are approximated by Legendre series truncated to a certain order W, as Equation (4.11) shows,

Algorithm 4.2 Integration of a *cpdf* to get a *ccdf*

Data:
(1) maximum order of *cpdf*: W;
(2) coefficients array of *cpdf*: $c[0, 1, \ldots, W]$.
Result: coefficients array of cumulative distribution function: $d[0, 1, \ldots, W+1]$.
// Initialize the coefficients
for $w = 0$ to $W + 1$ **do**
    $d[w] = 0$;
end for
// Update the coefficients according to Equation (4.25)
for $w = 1$ **to** W **do**
    $d[w+1] = d[w+1] + \frac{c[w]}{2w+1}$;
    $d[w-1] = d[w-1] - \frac{c[w]}{2w+1}$;
end for
$d[0] = d[0] + \frac{1}{2}$;
$d[1] = d[1] + \frac{1}{2}$;
END OF ALGORITHM

the number of the different coefficients is $(W+1)^{N+1}$, where N is the number of data points. Even the Legendre series are approximated by truncated series, where the sum of orders of different variables is not greater than W, which is the form adopted in Mustapha and Dimitrakopoulos [19]. The number of the different coefficients is still as big as $\sum_{w=0}^{W} \binom{N+w}{w}$ for a single data event. Although this computational complexity can be reduced by discarding some terms which are regarded as negligible, it should be noted that this simplification may lead to a loss of accuracy.

From Equations (4.15) and (4.19), it can be seen that all of the different coefficients introduced by the explicit expansion of Legendre series are reduced to a calculation of the function $\prod_{i=1}^{N} X_t(z_i)$. There are only NW computations of Legendre polynomials and a few products and additions included in the calculation of the function $\prod_{i=1}^{N} X_t(\zeta_{t,i})$ for each data event encountered in the training image. It should be noted that the computational time still depends on the number of the replicates encountered in the training image, as well as the maximal order of Legendre polynomials and the number of conditioning in the neighborhood. However, the computational cost regarding the above-mentioned parameters is significantly reduced, as opposed to computing the large number of coefficients in the previous version of high-order simulation.

4.5 Implementation

The implementation is relatively straightforward in terms of the above algorithms estimating the *cpdf* and *ccdf* according to the framework of sequential simulation. However, a method

is proposed in this section to deal with the replicates, aiming to reduce the conflicts of spatial statistics between the sample data and the training image. The main idea of the method is to deliberately select replicates which are similar to the conditioning data within a certain range according to some measure of similarity. The reason is that the conditional probability distribution is a one-dimensional intercept from the multivariate joint probability distribution, and therefore the replicates that are close to the conditioning data are more relevant to estimate this one-dimensional local probability distribution.

For every node to be simulated in sequential simulation, a local neighborhood is defined to search for conditional data from both the sample data and the simulation grid. The locations of these conditional data together with the center node to be simulated constitute a geometry template. Given a training image, replicates of the geometry template can always be found from the training image as long as the searching neighborhood is inside of the training image's extent. In the present work, the measure of similarity between the replicates and the data event is set to be the average square Euclidean distance between the replicates and the conditioning data, and the threshold is set as the variance of the sample data. The replicate will be selected in the estimation of *cpdf* if the distance between the replicate and the conditioning data is less than the variance of the sample data. In addition, when there are few replicates that can be found from the training image due to the conflicts between the sample and the training image, some tolerances are given to the shape of the geometry template so that similar replicates can be found. Figure 4.2 shows a general way to search the candidate points associated with a certain vector in a spatial template. The parameter θ is the angle tolerance of the candidate point's deviation from the original vector in the template, and Δh and b are the tolerances in the lag and bandwidth, respectively. Possible candidate points are taken from the shadowed area, and the point that has the closest property to the ending node of the original vector in the template is selected. To maintain the consistency of the geometry configuration, an inner part of the template is specified such that the relative locations to the center node inside the inner part remains unchanged. In other words, only the nodes further away from the center node are allowed to have the ability to change locations. This strategy gives more flexibility to manipulate the geometry configuration of the replicates.

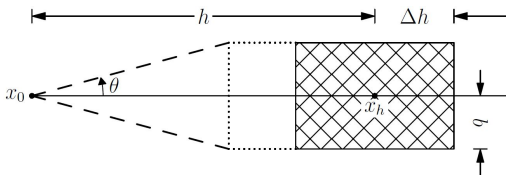


Figure 4.2 Finding approximate replicates from the training image with the tolerances of the original geometry template

The main procedure of the high-order simulation approach can be summarized in the following steps:

1. Read the sample data and training image into memory. In order to apply the multivariate expansion of Legendre polynomials, the property values of the samples or training image are scaled to the interval $[-1, 1]$ through a linear transformation.

2. Specify dimensions of a certain neighborhood for searching the conditional data and other parameters such as the minimum or maximum number of the conditional data. The geometry of the local template totally depends on the locations of the conditional data. In the present work, a rectangular shape neighborhood was used and searching policy was applied to find the closest points to the center. Nevertheless, the shape of the neighborhood and the searching policy can be manipulated to further control the spatial configuration of the template.

3. Set the lag tolerance, angle tolerance and bandwidth tolerance to enable searching approximate replicates from the training image (see Figure 4.2).

4. Generate a random sequence on the indices of the simulation grid to create a random visiting path.

5. According to the predefined visiting path, sequentially pick one node at a time for the simulation. If the property value is already known (copied from the hard data), then continue to choose another single node until the property value is not assigned. The conditioning data are searched inside the neighborhood centered on the chosen node by the previously specified searching policy from both the hard data and the simulated nodes.

6. A local spatial template is determined by the data and the center node for later simulation. This spatial template is then used to find similar replicates from the training image according to the parameters set in Steps 2 and 3. If the number of approximated replicates is not adequate for statistical inference, then drop the furthest node to the center node and repeat until the minimum number of conditioning data is reached.

7. The local *ccdf* is estimated from the replicates using the algorithms elaborated in Section 4.4. A random value is drawn from the local *ccdf* using the Monte Carlo method and set as the property value of the node to be simulated.

8. Repeat from Step 5 until all the nodes in the random path are visited.

4.6 Examples and Comparisons

The data used in this paper are extracted from the Stanford V reservoir data set [129]. A horizontal section serving as the exhaustive image is taken from the Stanford V reservoir model of porosity in a square grid with 100×100 pixels (cells of size 100 meters). As seen from the exhausting image in Figure 4.3, porosity values are distributed as several channels that can be distinguished from the background. For the examples and comparisons presented in the next sections, 200 data points are randomly sampled from the selected exhaustive image to serve as the sample data set and are displayed in Figure 4.4. Applying the proposed high-order approach, the selected data is used to simulate the exhaustive image in two different ways, so as to show the sensitivity of the approach to the chosen TI. Accordingly, in Example 1, the exhaustive data is used as the TI; then, in Example 2, the TI is selected from a diferrent section of the Stanford V reservoir data set than the exhaustive image. The second TI is shown in Figure 4.5 and has different spatial patterns than those in the exhaustive image. In addition, a comparison of the proposed algorithm to the well-established *mps* method *filtersim* [66] is presented. In each of the realizations using the high-order simulation algorithm, a window of size 15×20 in terms of cell size is used as the search template. The tolerance angle for searching is set to 15 degrees, the lag tolerance to 2 and the bandwidth to 1. These parameters are chosen from the calculation of experimental variograms [26]. The minimum number of conditioning data is 6 and the maximum number is 12, while 5 previously simulated values are used. The maximum order of Legendre polynomials is set to 10. For the realizations generated with *filtersim*, the searching template is 15×21 with an inner patch of size 7×7 and a multiple grid level of 3, while replicates are classified into different categories according to their filter scores. For further details on *filtersim*, the reader is referred to Zhang et al. [66].

4.6.1 Example 1

This example generates simulations using the 200 samples shown in Figure 4.4 and the exhaustive image in Figure 4.3 as the TI. In this case, there are no conflicts between the available data and the training image. Figure 4.6 shows one realization from the high-order simulation and another from *filtersim*, respectively. From visual comparison with the exhaustive image, the realization from the high-order simulation reproduces better the channels of the original image. To demonstrate the reproduction of the distribution and second-order spatial statistics of simulation results, 10 different realizations for each method are generated. The histograms of the realizations are displayed in the Figure 4.7 and related variograms are displayed in Figure 4.8. Both simulation methods reproduce well the bimodal shape

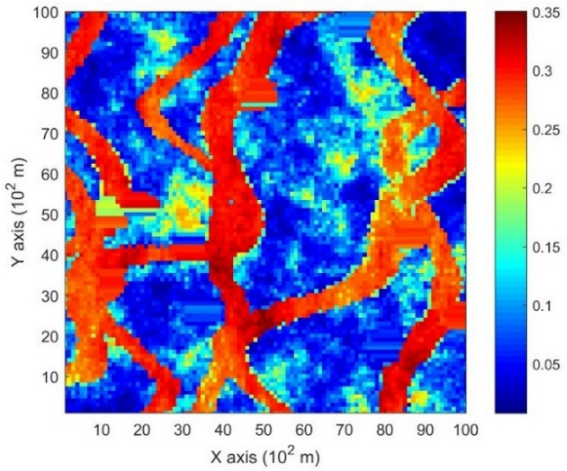


Figure 4.3 A horizontal section from reservoir's porosity values with sinuous connectivity

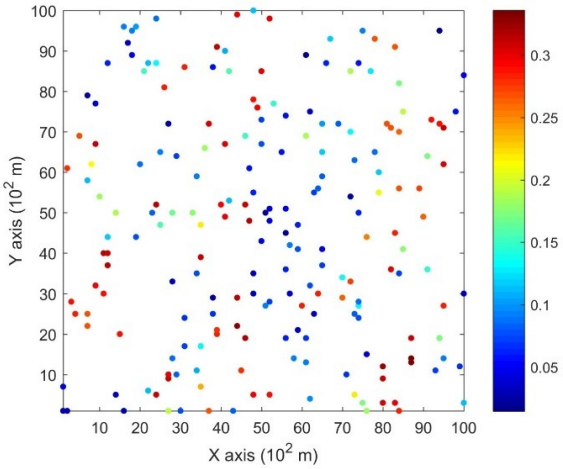


Figure 4.4 Data points sampled from the exhaustive image (containing 200 points, or 2% of the total data)

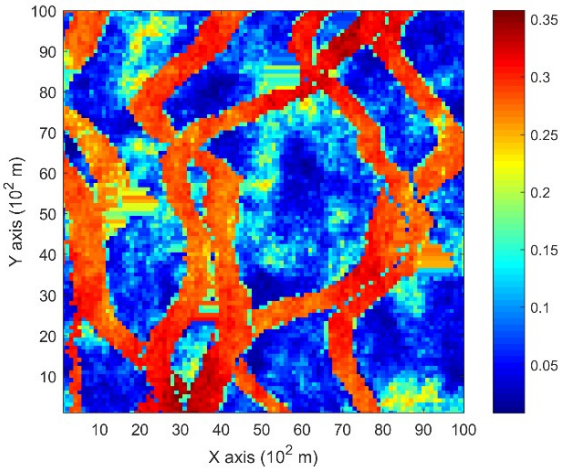
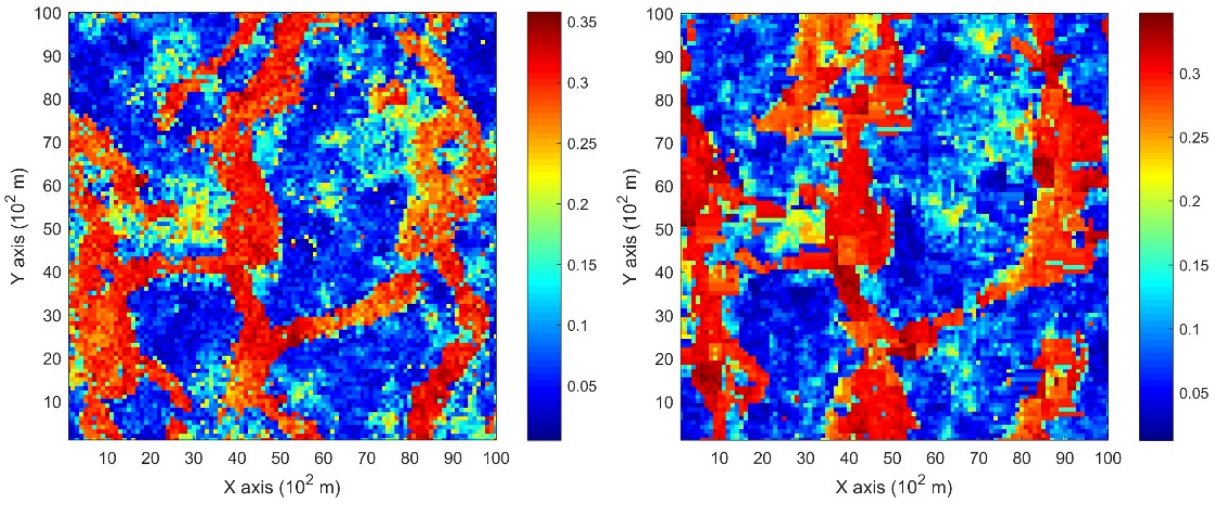


Figure 4.5 Training image that is different from the exhaustive data

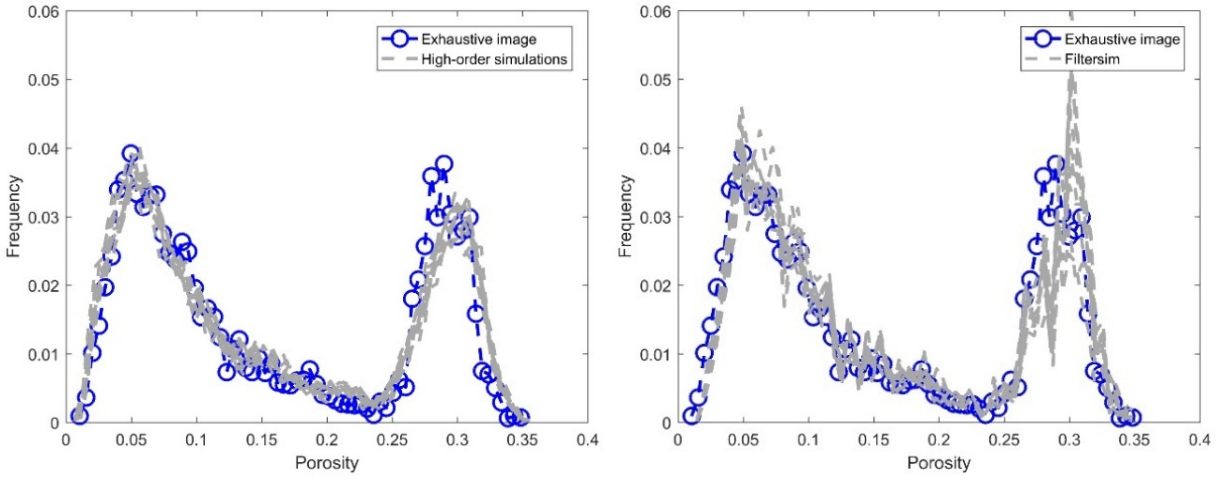
in the histograms; however, in general, high-order simulations show better reproduction in the proportions of porosity values. High-order simulation methods also reproduce well the variograms in the X-direction or Y-direction, while the variograms from the *filtersim* simulations demonstrate larger fluctuations and have notable deviations from the variogram of the exhaustive data in the Y-direction. For a comparison of the high-order spatial statistics of simulation results to the original data in the two different settings, the third-order cumulant maps are generated by the HOSC program [91]. This program uses a template with two directions in X-axis and Y-axis, and the number of lags are 70 with lag size as 1, which are

displayed in Figure 4.9. In comparison to the third-order cumulant map of the exhaustive image, the high-order simulation performs better in the reproduction of the high-order statistics, although both simulation methods have reasonable similarity in terms of the third-order cumulant map, as there are no conflicts between the sample data and the training image in this case.



(a) (b)

Figure 4.6 Simulations with 200 sample data using the exhaustive data as the training image. one realization from (a) high-order simulation and (b) *filtersim*



(a) (b)

Figure 4.7 Reproduction of histograms of 10 realizations with 200 sample data using the exhaustive data as the training image: (a) and (b) correspond to 10 realizations from the high-order simulation and *filtersim*, respectively

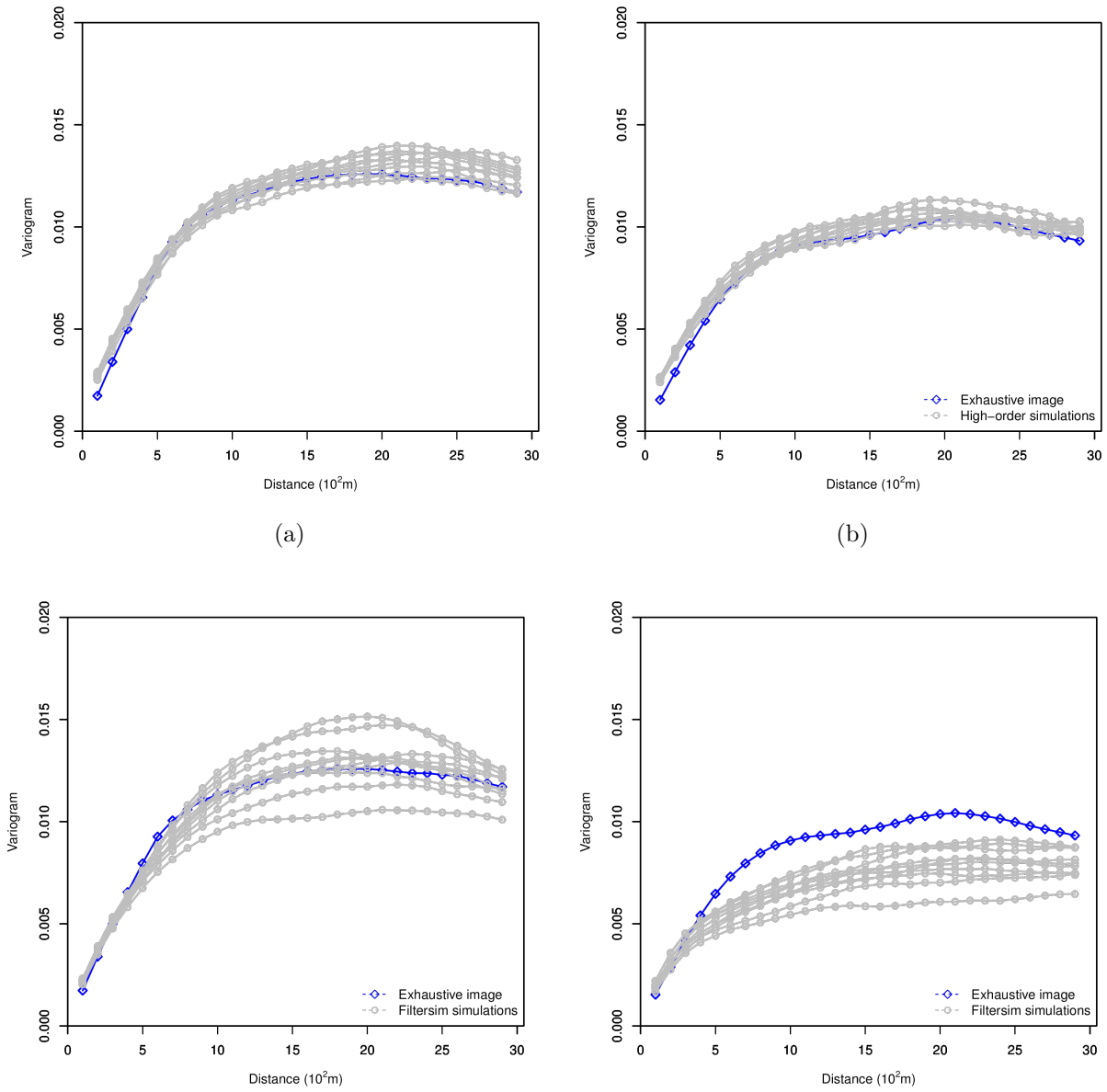


Figure 4.8 Reproduction of the variograms of 10 realizations with 200 sample data using the exhaustive data as the training image from high-order simulation and *filtersim*, respectively. (a) Reproduction of variograms of high-order simulations in the X direction; (b) Reproduction of variograms of high-order simulations in the Y-direction; (c) Reproduction of variograms of *filtersim* simulations in the X-direction; (d) Reproduction of variograms of *filtersim* simulations in the Y-direction

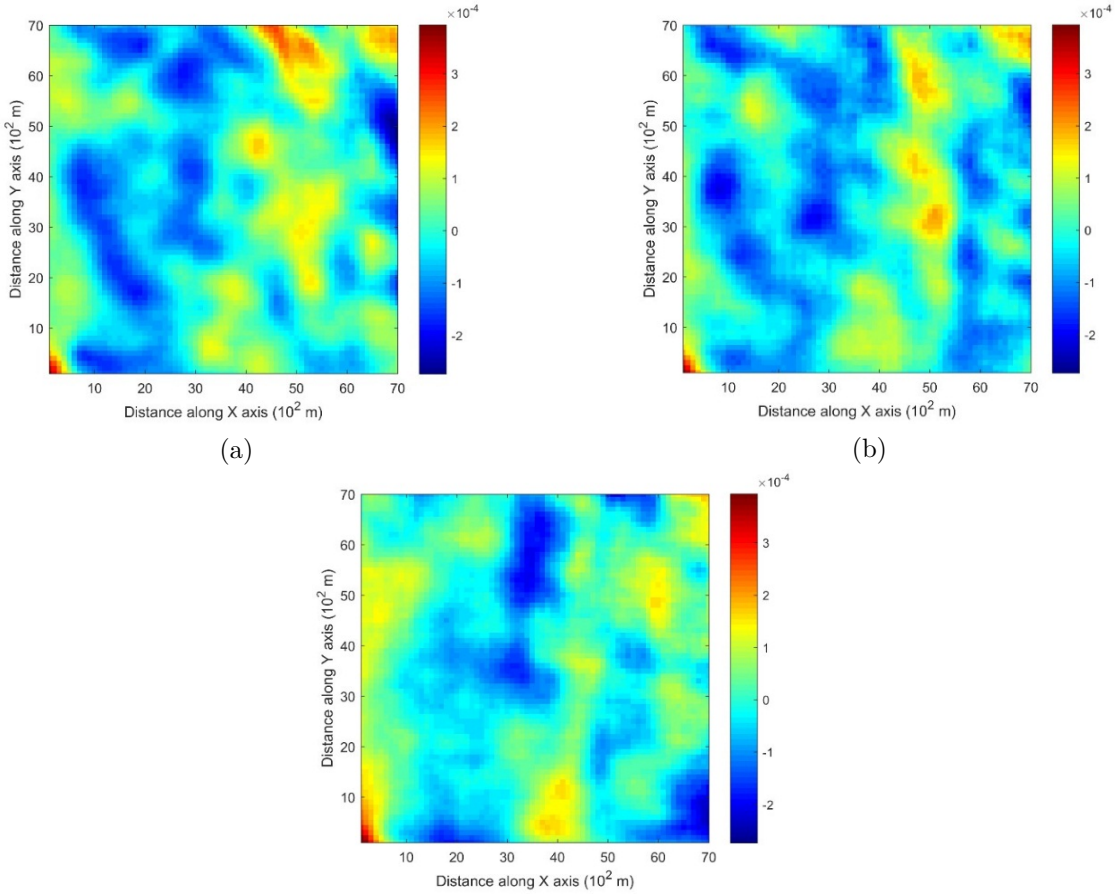


Figure 4.9 Comparing third-order cumulant maps of realizations with 200 sample data using the exhaustive data as the TI from the high-order simulation and *filtersim*, respectively. a Third-order cumulant map of the exhaustive image. b Third-order cumulant map of one realization from the high-order simulation. c Third-order cumulant map of one realization from *filtersim*

4.6.2 Example 2

In this setting, the simulations are conducted with the same conditioning data; however, the TI is different from the exhaustive data. Figure 4.10 shows one realization from the high-order simulation and one for *filtersim*. Clearly, there are conflicts between the spatial statistics of the sample data and the TI, which are key factors affecting the results of the simulations. As expected, the reproduction of the spatial patterns is worse when compared to the results from the simulations in the previous example. Nevertheless, the realization from high-order simulation method still maintains the spatial structures of the original exhaustive data. As shown in Figure 4.11, the 10 realizations of the high-order simulation match the histogram of the exhaustive image very well. By contrast, the 10 realizations of *filtersim* mismatched the

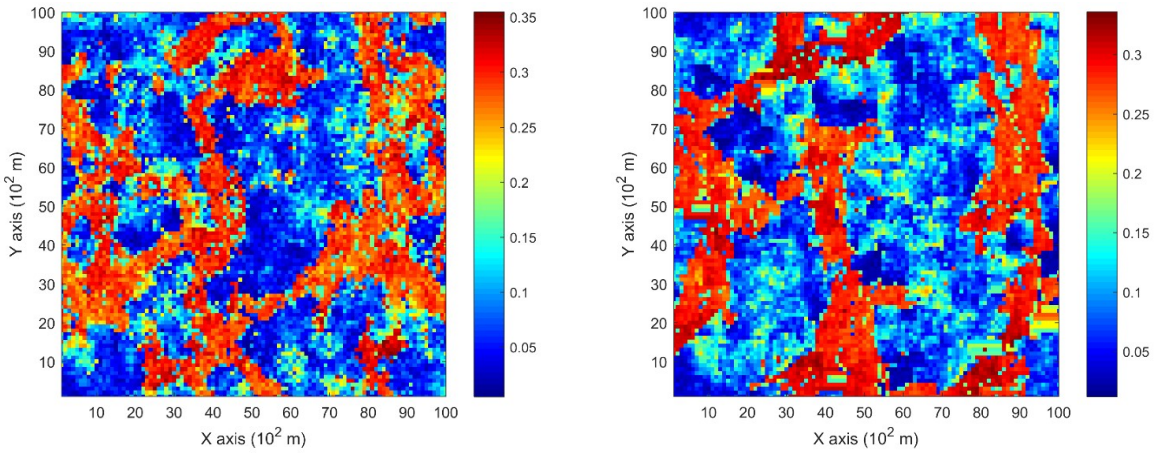
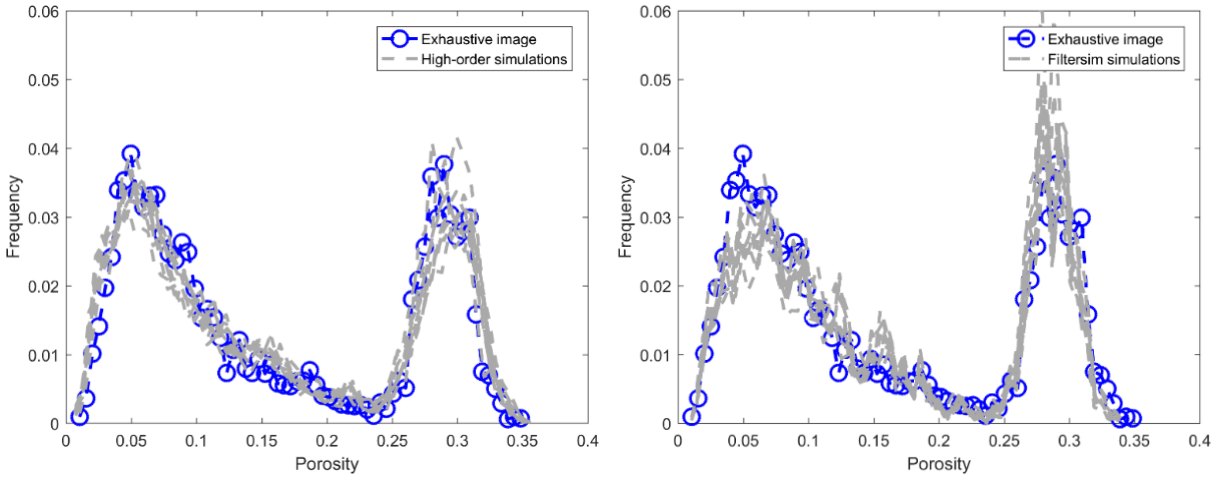


Figure 4.10 Simulations with 200 sample data using a separate training image different from the exhaustive data. one realization from (a) high-order simulation and (b) *filtersim*

exhaustive image in some part of the proportions. From the comparison shown in Figure 4.12, the high-order simulation performs better than *filtersim* in reproducing the variograms of the exhaustive image as well, although there is a minor deviation in the Y-direction. In order to demonstrate the impact of the conflicts between the sample data and training image during the simulations, Figure 4.13 (a)–(e) shows the third-order cumulant maps corresponding to the exhaustive image, the sample data, the training image and one realization of high-order simulation and *filtersim*. The parameter settings to generate the cumulant maps for the grid data are the same as used in Figure 4.9, whereas the lag size is set to 5 grid cells, with the lag tolerance being set to 1 grid cell and the angle tolerance being set to 15 degrees for generating the cumulant map of the sample data. As the sample data is too sparse to compute the cumulant map at the same scale as the exhaustive image, some smoothing has been applied to the cumulant map of the sample data for the purpose of visualization. The third-order cumulant map of the realization from the high-order simulation maintains the main structures of the exhaustive data. On the other hand, the third-order cumulant map of the realization from *fitlersim* resembles the cumulant map of the TI, which differs from the cumulant map of the exhaustive image. This implies that the high-order simulation is primarily data-driven, whereas the *filtersim* method is TI-driven. This result can be explained by the fact that the high-order simulation seeks to find replicates that comply to the statistical configuration of the conditioning data from the training image, and the values of nodes to be simulated are drawn from the related local probability distribution. By contrast, the *filtersim* method is TI-driven, which means that the values of nodes to be simulated comes directly from the paste of certain replicates from the training image, which is patch-based instead of node-by-node

as in the high-order simulation. In particular, the impact of the conditioning data is more important for capturing the large-scale spatial structures in the early stage of the high-order simulation. For instance, Figure 4.13 (b) shows the cumulant map of the sample data, and the resolution of the map is much coarser than the exhaustive data. This map shows some distortion when representing the third-order statistics of the exhaustive image due to the sparsity of the data. However, the spatial structures of the limited sample data control the spatial statistics of the results from the high-order simulation. In general, the results in this case study show that the proposed high-order simulation algorithm can reasonably reproduce the overall probability distribution, the second-order statistics and the higher-order statistical features (such as spatial cumulants), as the statistical conflicts between the sample data and the training image are not severe.



(a) (b)

Figure 4.11 Reproduction of histograms of 10 realizations with 200 sample data using the training image different from the exhaustive data: (a) and (b) correspond to 10 realizations from high-order simulation and *filtersim*, respectively

4.6.3 Parameter sensitivity testing

Most parameters in the current implementation of high-order stochastic simulation method are experimental choices. Amongst all the parameters encountered in the current implementation, some follow common practices in the parameter selection for conventional geostatistical simulations, such as the size of the search window, the lag and angle tolerance. Additionally, in the high-order simulation method presented here, the number of conditioning data corresponding to a certain template needs more consideration, as it determines the dimension of the local probability distribution. In the current implementation, the number

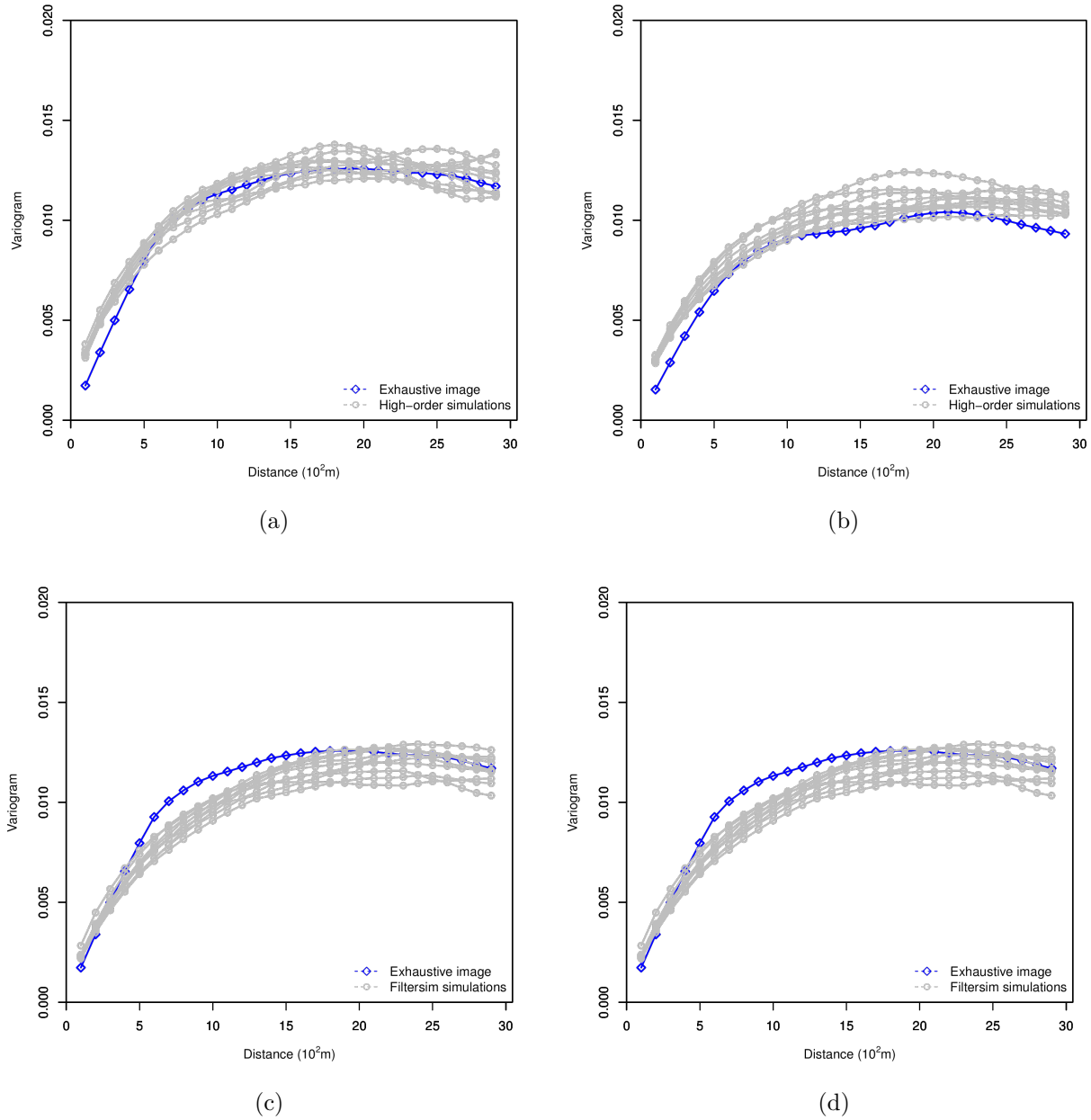


Figure 4.12 Variograms of 10 realizations with 200 sample data using the training image different from the exhaustive data from the high-order simulation and *filtersim*, respectively

of the conditioning data is limited for two important reasons. First, the limited number of conditioning data reduces the computational time needed to estimate the *cpdf*. Second, the method resembles the so-called multiple grid strategy [57] applied in many multi-point simulation methods in order to maintain both large and small-scale spatial structures. In the early stage of the simulation process, the neighborhoods are more likely to capture large-

scale patterns, since the known data are sparse. The neighborhoods gradually correspond to finer-scale patterns as the simulation continues and more known data are generated. A similar search strategy has also been applied and discussed in [60]. The maximum order of the polynomials is another parameter of importance in the high-order simulation, since it affects the precision of the approximation of *cpdf* by a truncated Legendre polynomial series. Theoretically, the coefficients in the Legendre polynomial series decay exponentially, and in general much faster than in Taylor series [130, 131]. The numerical results in [130] show that Legendre polynomial series with six non-zero coefficients (order 10 and 11 in their examples) are highly accurate approximations to the targets. The numerical testing to approximate a probability distribution regarding the order of Legendre polynomial series has also been investigated in [18] and led to similar results. However, it should be noted that the above tests are conducted for the approximation of a determined function, whereas for the approximation of the probability density function, there is also the impact from the limitation of the number of replicates. Depending on different data sets, Legendre polynomial series with order from 6 to 20 should be a reasonable range to select.

For validation and sensitivity analysis, further tests are conducted specifically to demonstrate the impacts of the number of conditioning data and the maximum order of Legendre polynomial series. In order to restrict the effects of the conflicting statistics between the TI and the sample data, the same data set from Example 1 is used to evaluate the sensitivity of the related parameters. The experiments are taken for each individual parameter without considering the possible dependencies between them. In all the experiments, the parameters not being tested remain the same as in Example 1. Furthermore, the random seed used to generate the visiting path is also fixed for all the simulations in the experiments, so that the impact of the different visiting path is excluded. Figures 4.14 and 4.15 depict the realizations of the high-order simulation with different neighborhood sizes and their corresponding third-order cumulant maps. In addition, Figures 4.16 and 4.17 show the realizations of high-order simulation with respect to the order of the Legendre polynomial series in order to approximate the *cpdf*s (as well as their corresponding third-order cumulant maps). From the results, it can be seen that both the size of the neighborhood and the maximum order of the polynomials have considerable impacts on the high-order simulation results. In particular, using a small size of the neighborhood of 6 grid cells or a Legendre polynomial order less than 6 results in a poor reproduction of the spatial patterns as well as the cumulant maps. However, when the size of the neighborhood increases to more than 12 or the order of polynomials is greater than 10, the differences become trivial. Although the testing is for a specific data set, and the size neighborhood should be larger in 3D space than 2D space, it can be expected that a similar sensitivity analysis can be applied to choose the appropriate parameters on a

case-by-case basis.

4.7 Conclusions

The main contributions of the paper are as follows. Firstly, starting from the high-order simulation method based on Legendre polynomial series, a new computational model in the form of a unified empirical function is developed to approximate the *cpdf*. The computational model leads to an estimation of *cpdf* without calculating the high-order spatial cumulants or moments term by term. As a consequence, it not only greatly reduces the computational requirements but also provides a more accurate approximation of *cpdf* through Legendre polynomial series in comparison to the previous high-order simulation algorithm based on Legendre cumulants. Secondly, two new algorithms to derive *cpdf* and *ccdf* based on the above computational model are developed; they both use the properties of Legendre polynomials to simplify the computation and avoid an explicit expansion of a multivariate Legendre series. Lastly, the spatial template used in the current high-order simulation method is dynamically changing with the computation of the probability distribution in real time without storing data events. In addition, a flexible strategy to search replicates from the training image is proposed and implemented to deal with the conflicts between the statistics of the sample data and the training image.

Tests show the capacity of the proposed algorithm to reproduce complex geological patterns, and, in addition, that both the overall distribution and the high-order spatial statistics of the data are reproduced by the high-order simulations. Comparing the results of the high-order simulation in different cases with those of *filtersim*, the high-order simulation outperforms in the reproduction of high-order spatial statistics. This result becomes more notable in cases where there are conflicts in the spatial statistics between the sample data and the training image. This demonstrates that the high-order simulation has a more data-driven nature, whereas the *filtersim* is more TI-driven. Although the computational cost is significantly reduced (depending on the size of the training image, the number of neighborhood and the maximum order of Legendre polynomial series), the simulation is still slower than the *filtersim* method. However, since the computations of the *cpdf* is carried out on each replicate with the same type of calculation, the procedure could be parallelized so that the simulation can be further accelerated through parallelization techniques, such as GPU programming. It also should be noted that the approximation of *cpdf*s by Legendre series or any kind of polynomial series may generate problems of non-positive probability densities; further research is needed to address the issue.

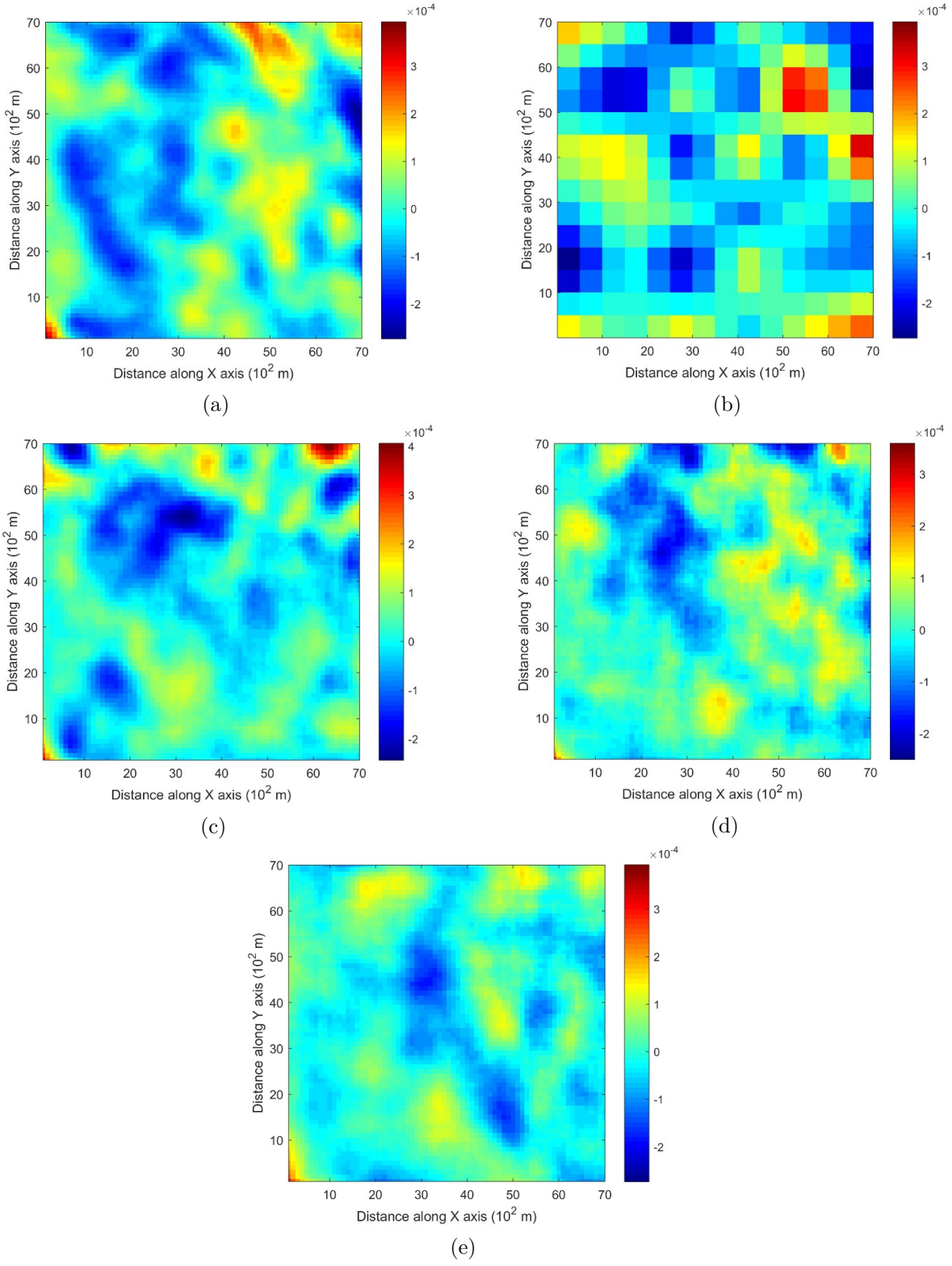


Figure 4.13 Third-order cumulant maps of (a) exhaustive image, (b) sample data, (c) TI, (d) high-order simulation and (e) *filtersim*

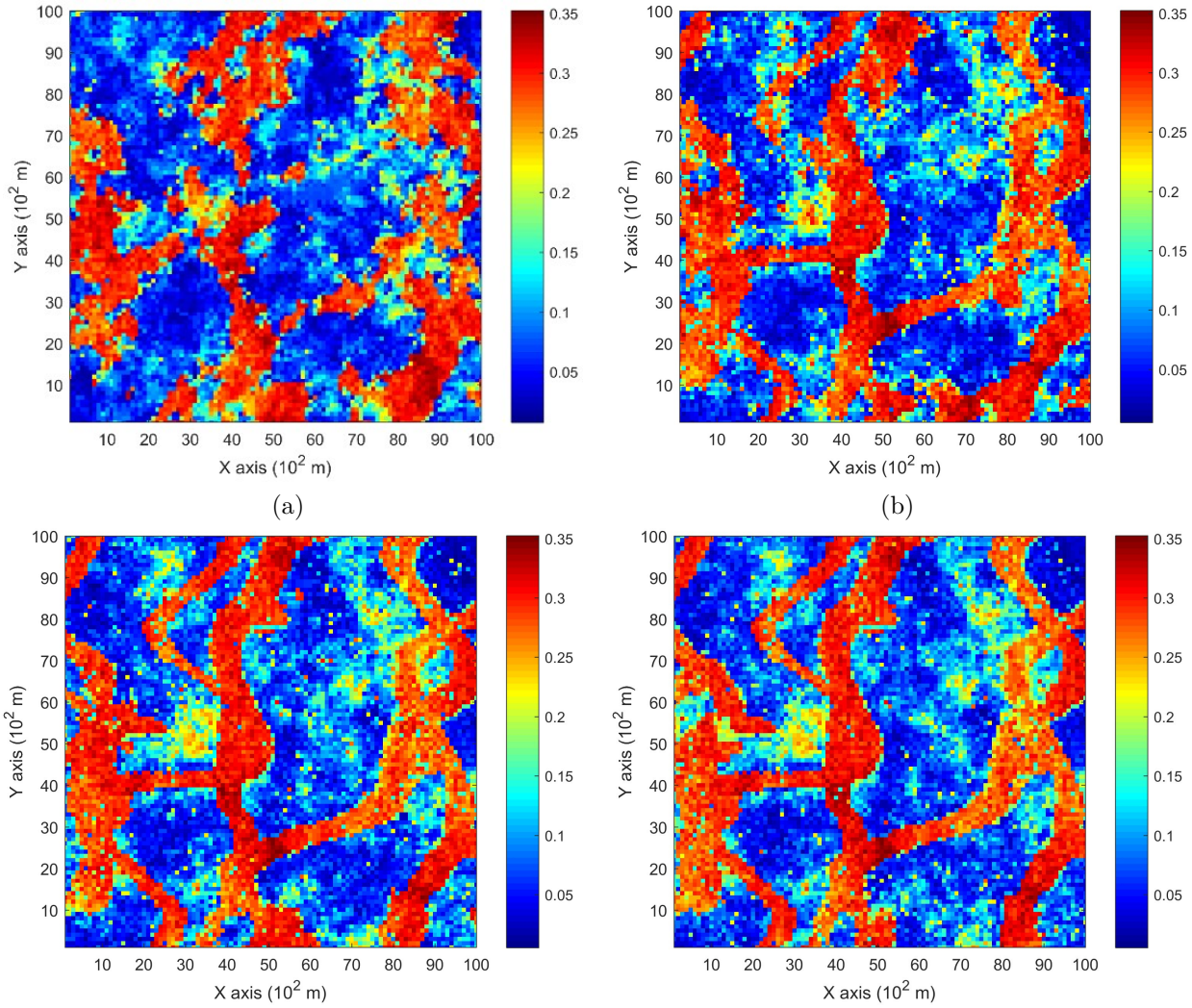


Figure 4.14 Comparing the realizations of high-order simulation by applying different local neighborhood size, with 200 sample data using the exhaustive data as the training image. The maximum order of Legendre polynomials to approximate the *cpdf*s is 10 for all the realizations. Realizations with neighborhood of (a) 6 conditioning data, (b) 12 conditioning data, (c) 20 conditioning data, and (d) 30 conditioning data

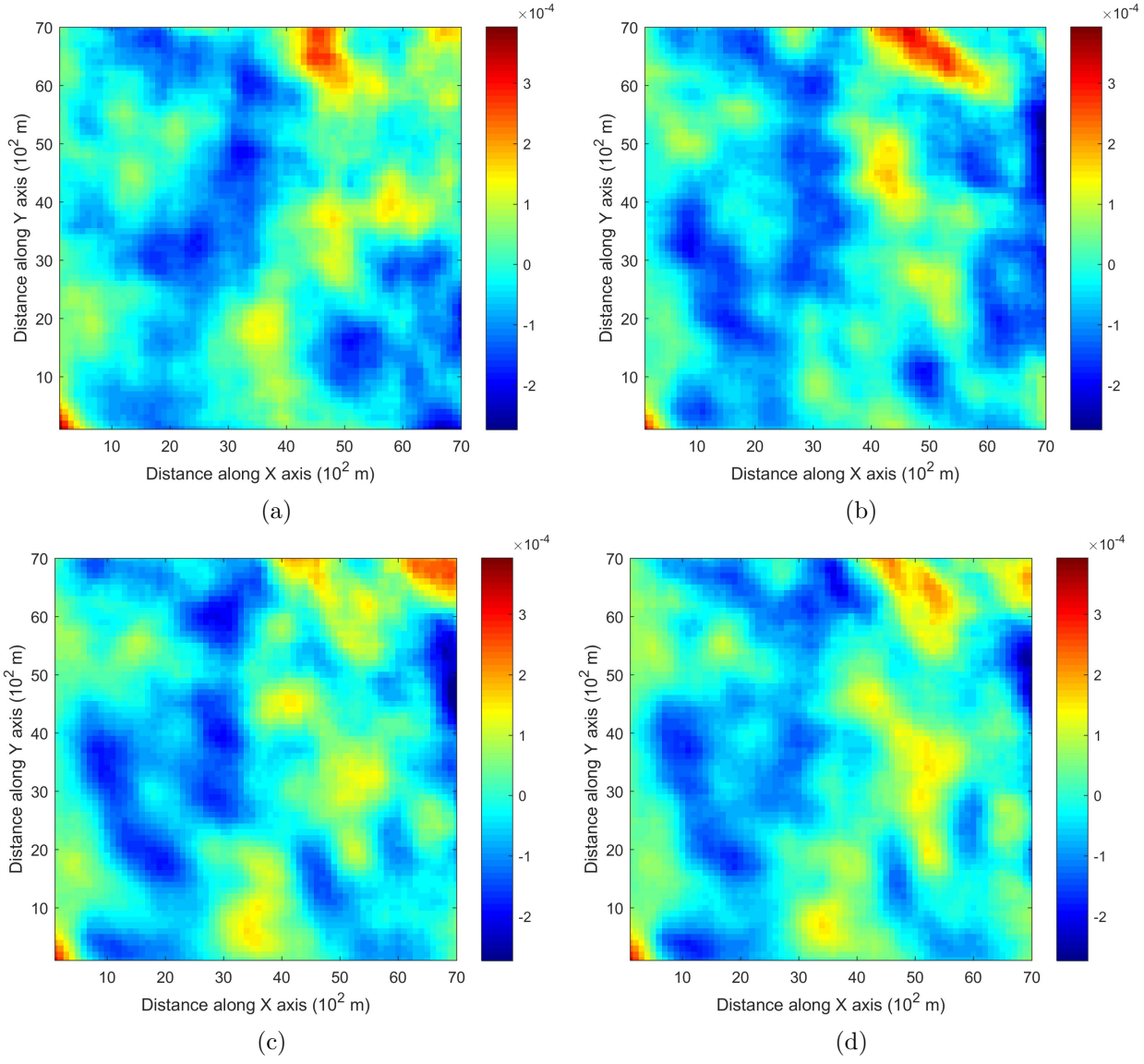


Figure 4.15 Comparing the third-order cumulant maps of the realizations of the high-order simulation by applying different local neighborhood size, with 200 sample data using the exhaustive data as the training image. The maximum order of Legendre polynomials to approximate the *cpdf*s is 10 for all the realizations. Third-order cumulant maps of one realization with neighborhood of (a) 6 conditioning data, (b) 12 conditioning data, (c) 20 conditioning data, and (d) 30 conditioning data

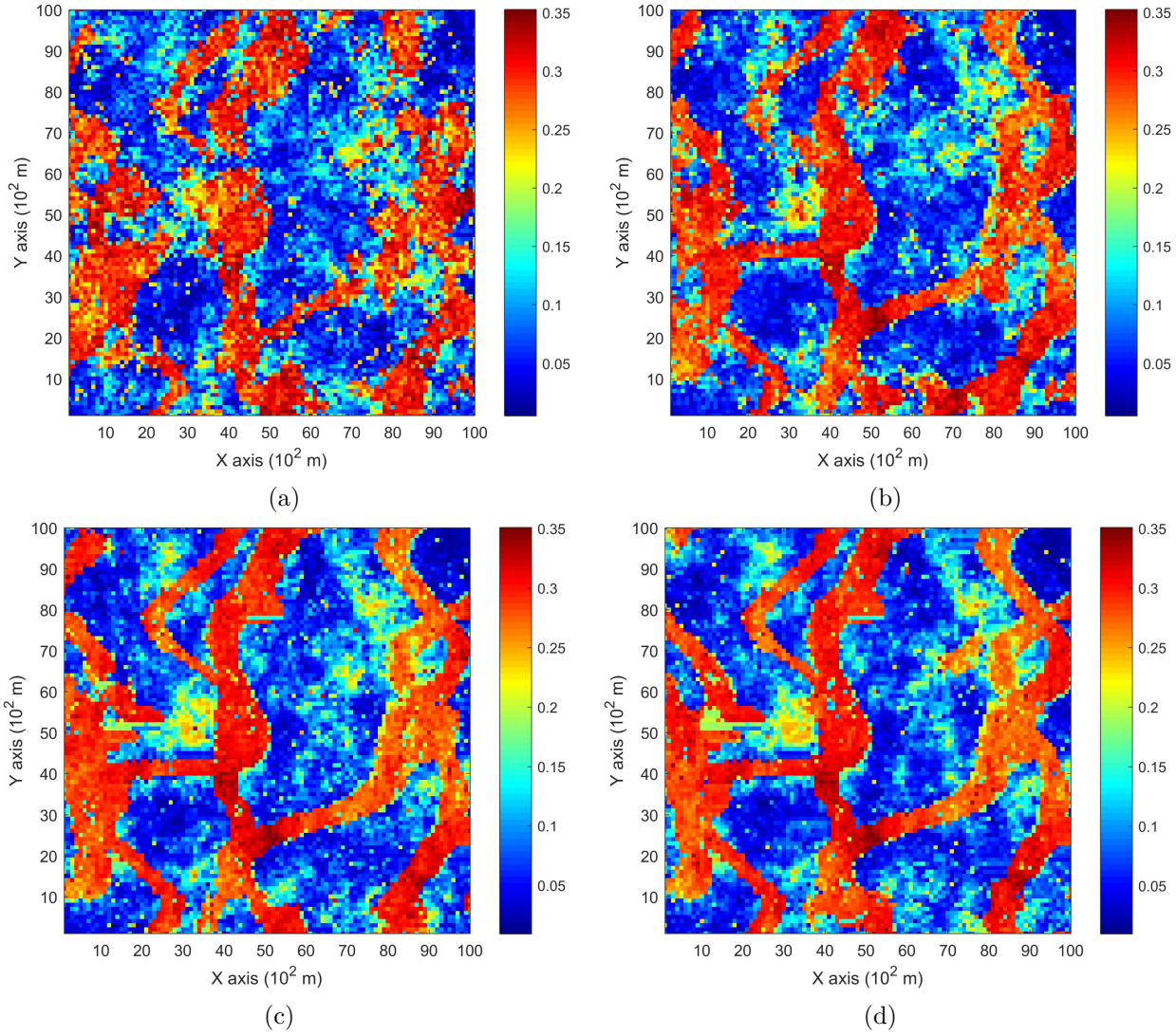


Figure 4.16 Comparing the realizations of the high-order simulation by applying the different order of truncated Legendre polynomial series, with 200 sample data using the exhaustive data as the training image. The number of conditioning data in the local neighborhood is 12 for all the realizations. Realizations of the high-order simulation by approximating *cpdf* with Legendre polynomial series up to (a) order 6, (b) order 10, (c) order 20, and (d) order 30

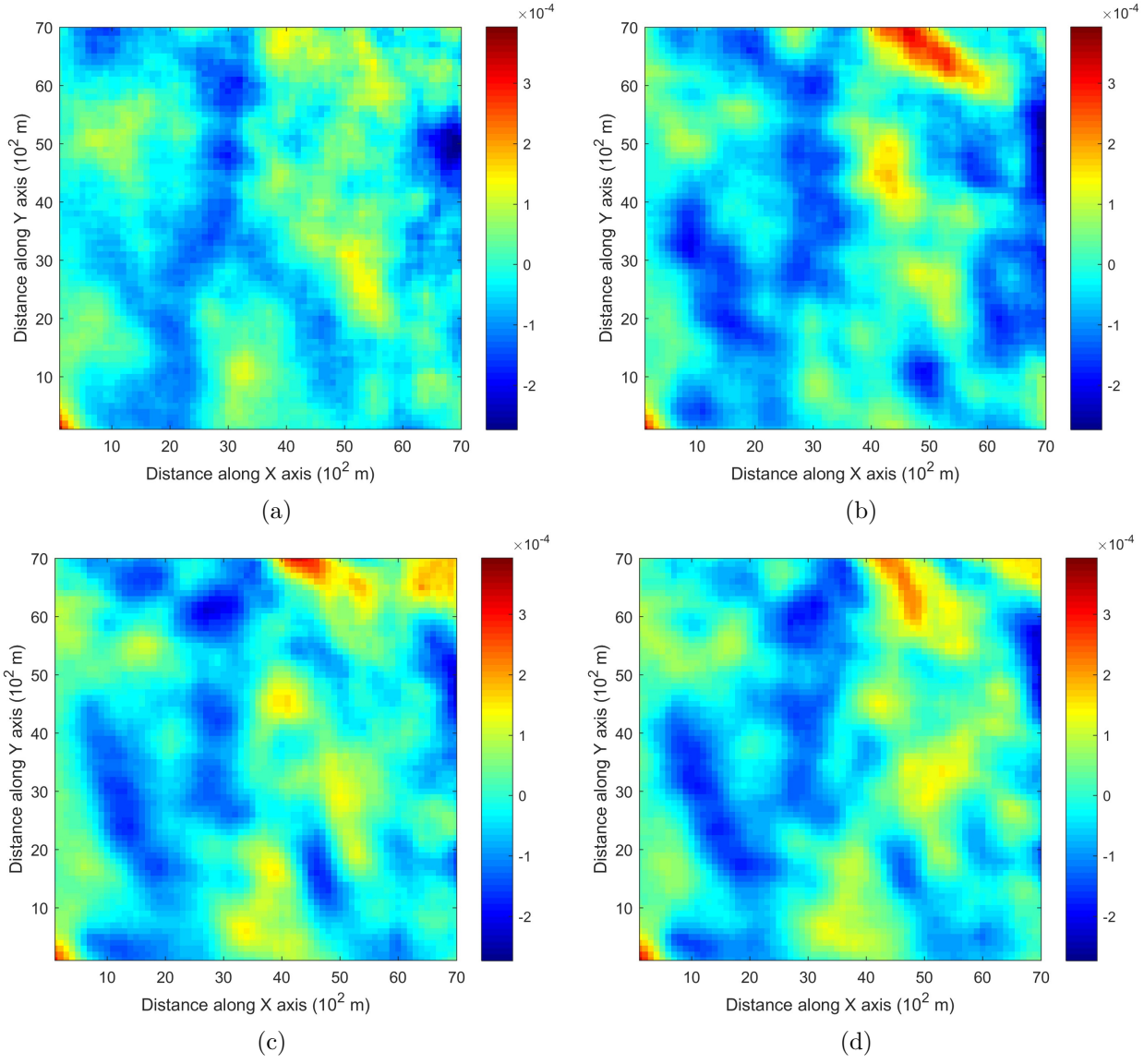


Figure 4.17 Comparing third-order cumulant maps of the realizations of the high-order simulation by applying the different order of truncated Legendre polynomial series, with 200 sample data using the exhaustive data as the training image. The number of conditioning data in the local neighborhood is 12 for all realizations. Third-order cumulant map of one realization of the high-order simulation by approximating *cpdf* with Legendre polynomial series up to (a) order 6, (b) order 10, (c) order 20, and (d) order 30

CHAPTER 5 ARTICLE 2: HIGH-ORDER SEQUENTIAL SIMULATION VIA STATISTICAL LEARNING IN REPRODUCING KERNEL HILBERT SPACE

Abstract: The present work proposes a new high-order simulation framework based on statistical learning. The training data consist of the sample data together with a training image and the learning target is the underlying random field model of spatial attributes of interest. The learning process attempts to find a model with expected high-order spatial statistics that coincide with those observed in the available data, while the learning problem is approached within the statistical learning framework in a reproducing kernel Hilbert space (RKHS). More specifically, the required RKHS is constructed via a spatial Legendre moment (SLM) reproducing kernel that systematically incorporates the high-order spatial statistics. The target distributions of the random field are mapped into the SLM-RKHS to start the learning process, where solutions of the random field model amount to solving a quadratic programming problem. Case studies with a known data set in different initial settings show that sequential simulation under the new framework reproduces the high-order spatial statistics of the available data and resolves the potential conflicts between the training image and the sample data. This is due to the characteristics of the spatial Legendre moment kernel and the generalization capability of the proposed statistical learning framework. A three-dimensional case study at a gold deposit shows practical aspects of the proposed method in real-life applications.

Keywords: Stochastic simulation, High-order spatial statistics, Statistical learning; Reproducing kernel, Multipoint simulation

5.1 Introduction

Stochastic simulations are used to quantify the spatial uncertainty in earth science or engineering applications. Since the early 1990s, the so-termed multipoint statistical simulation (MPS) methods [12, 55, 57] were first proposed to overcome the limitation of the second-order simulation approaches in reproducing the complex spatial patterns encountered in natural phenomena. Instead of using a theoretical variogram/covariance model, as is the case with conventional two-point geostatistical simulations, the MPS methods consider that the so-

Published: Yao L, Dimitrakopoulos R, Gamache M (2019) High-order sequential simulation via statistical learning in reproducing kernel Hilbert space. Math Geosci. doi:10.1007/s11004-019-09843-3

called training image (TI) contains the prior information of the spatial statistics or patterns of the attribute to be simulated. A spatial template is defined as a geometrical configuration of the relative locations among the multiple points, regardless of the coordinates. The known data within the spatial template at a certain location on the simulation grid acts as the conditioning data in the simulation and is termed a data event. Over the past decade, several state-of-the-art MPS algorithms have been proposed to improve the efficiency and reproduction of the curvilinear features [56, 122].

An inherent limitation of the MPS algorithms is that the high-order spatial statistics of the available data are not systematically considered and are partly integrated in ad-hoc ways. This issue becomes more prominent when the spatial statistics of the TI and the sample data are different, leading to realizations conflicting with the spatial statistics of the sample data, especially when the latter data is relatively dense as is the case in mining applications [125, 126]. As an alternative, high-order simulation methods are proposed to model a random field without any presumption of its probability distribution, and high-order spatial statistics are systematically incorporated in the model [17, 18, 95]. The first algorithm of high-order simulation, HOSIM, approximates the probability density function (PDF) by the Legendre polynomial series through the so-called spatial cumulants [17–19]. Further developments of the high-order simulation paradigm include the simulation of spatially correlated variables [127] and the direct simulation at the block scale [132]. Most recently, Yao et al. [133] proposed a new computational model of high-order simulation as a unified empirical function, which avoids CPU-demanding computations of expansion coefficients. Furthermore, a kernel function can be derived from this model and will be used in the present work.

A common issue that runs across all of the above-mentioned high-order simulation methods is that the approximation of the PDF by orthogonal polynomials cannot be guaranteed to be positive. The sensitivity of high-order polynomials to the rounding errors near the endpoints of the approximation weakens the convergence of polynomial series to a stable analytic function, as discussed in Minniakhmetov et al. [128], who propose an approximation of the PDF using Legendre-like orthogonal splines as the basis functions, resulting in a significant improvement in numerical stability. As the deviation of the empirical statistics from the true expectation arises due to possible statistical conflicts between the sample data and the TI, the convergence of the approximation to the actual underlying PDF could be undermined. Under such a circumstance, a postprocessing step has to be introduced to correct the approximation. For example, the correction procedure through interpolation around the points of negative densities is applied in Mustapha and Dimitrakopoulos [18].

The present work proposes a new high-order simulation framework based on statistical learn-

ing [110, 134], which deliberately mitigates the statistical conflicts between the sample data and the TI, and also overcomes the limitation of approximating the PDF with the orthogonal expansion series. Statistical learning theory [110] develops a new learning paradigm to explore functional dependency from a given data set without relying on prior knowledge, which contrasts with the classical statistical methods that are based on parametric models. According to the learning paradigm, a target model needs to be learned from the available data set, which represents the training data. The so-called learning machine [110] is frequently given as a set of functions, from which a specific learning model is selected to approximate the target model according to certain criteria.

To interpret high-order simulation in terms of statistical learning, the training data are regarded as the available data from the sample data and/or the TI. The target model is the probability distribution related to the random field of the spatial attributes. The learning model is the approximated PDF of the target probability distribution, from which the realizations can be generated. The learning process for high-order simulation is driven by matching the expected high-order spatial statistics of the target probability distribution to the high-order spatial statistics observed from the available data. The matching of the high-order spatial statistics is the most challenging part and is approached herein by a learning process in a reproducing kernel Hilbert space (RKHS) [111]. A spatial Legendre moment (SLM) reproducing kernel is proposed to construct the specified RKHS (SLM-RKHS), such that the high-order spatial statistics are systematically incorporated in this Hilbert space for a certain probability distribution. The elements in the original data space are mapped into the SLM-RKHS, termed RKHS embedding [135–138]. In addition, the high-order spatial statistics of the available data are carried over to the domain after this RKHS embedding. Eventually, the statistical learning regarding high-order simulation leads to a convex optimization in SLM-RKHS where the solutions amount to solving a quadratic programming problem.

In the following sections, the general theory of kernel methods, including the reproducing kernel Hilbert space (RKHS) and RKHS embedding of probability distributions, are introduced. Section 5.2 describes the main workflow of high-order sequential simulation via statistical learning, and a spatial Legendre moment reproducing kernel is defined to construct the specific SLM-RKHS. Furthermore, this SLM-RKHS is decomposed to lower-dimensional subspaces, such that conditional probability density functions (CPDF) in the context of sequential simulation can be embedded into the corresponding subspaces. Subsequently, a high-order stochastic simulation method is presented as a learning process based on the embedding of the CPDF into the decomposed subspace of the SLM-RKHS. Next, the proposed simulation method is tested using a fully known data set. A case study at a gold deposit is

then presented to show the practical aspects of the proposed method. Conclusions follow.

5.2 Method

5.2.1 Overview of kernel space and embedding a probability distribution

In the general setting of kernel methods, a kernel space needs to be set up and associated with a predefined kernel function, and a feature mapping is defined to map an arbitrary element from the original data space into the kernel space. The related general concepts and theory are formalized in the followed subsections.

5.2.1.1 Reproducing kernel Hilbert space

A Hilbert space $\mathcal{H}$ is a vector space over a field endowed with an inner product [139]. For simplicity, the Hilbert space $\mathcal{H}$ over the set $\mathbb{R}$ of real numbers is considered here, and the inner product is defined as

$$\langle f, g\rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}, \quad \forall f, g \in \mathcal{H}.$$

The norm is defined as

$$\|f\|_{\mathcal{H}} = \langle f, f\rangle^{1/2}, \quad \forall f \in \mathcal{H}.$$

Other essential properties can be found in [139]. The concepts of reproducing kernel and positive definite function are from [116] with the modification of the range of kernel function to $\mathbb{R}$.

Reproducing kernel

Let $\mathbb{E}$ be a non-empty set and $\mathcal{H}$ be a Hilbert space of functions defined on $\mathbb{E}$. Then, a function $K : \mathbb{E} \times \mathbb{E} \to \mathbb{R}$ is a reproducing kernel of a Hilbert space $\mathcal{H}$ if and only if

(1) $\forall t \in \mathbb{E}, K(\cdot, t) \in \mathcal{H}$, and

(2) $\forall t \in \mathbb{E}, \forall f \in \mathcal{H}, \langle f, K(\cdot, t)\rangle = f(t)$.

The last condition is called "the reproducing property," because any function in $\mathcal{H}$ can be reproduced by its inner product with the kernel K. In addition, as a direct derivation of the above conditions, the reproducing kernel can be written as the inner product

$$K(s, t) = \langle K(\cdot, s), K(\cdot, t)\rangle, \quad \forall s, t \in \mathbb{E}.$$

Naturally, a Hilbert space in possession of a reproducing kernel is called a reproducing kernel Hilbert space. The feature map associated with an RKHS $\mathcal{H}$ with kernel K is defined as $\phi: \mathbb{E} \rightarrow \mathcal{H}$ such that $\langle \phi(s), \phi(t) \rangle = K(s,t)$. In fact, $\phi(t): \mathbb{E} \rightarrow \mathcal{H}, t \mapsto K(\cdot, t), \forall t \in \mathbb{E}$ satisfies such a definition as the feature map according to the reproducing property. This type of feature map is called a reproducing kernel map [111] or canonical feature map [140] and will be used in the present paper.

Positive definite function

A real-valued function $K: \mathbb{E} \times \mathbb{E} \rightarrow \mathbb{R}$ is positive definite if $\forall n \geq 1, \forall (a_1, \ldots, a_n) \in \mathbb{R}^n, \forall (x_1, \ldots, x_n) \in \mathbb{E}^n$, there is

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j K(x_i, x_j) \geq 0.$$

5.2.1.2 RKHS embedding of a probability distribution

The range of the feature mapping spans RKHS $\mathcal{H}$ by definition [111]. Thus, the feature mapping ϕ is crucial in embedding a data element into the RKHS $\mathcal{H}$. Accordingly, two mappings are important to embed a probability distribution into the RKHS $\mathcal{H}$ [135]

$$\mu[p] = \mathrm{E}_{x \sim p}[\phi(x)], \tag{5.1}$$

and

$$\mu[X] = \frac{1}{M} \sum_{i=1}^{M} \phi(X_i), \tag{5.2}$$

where the first equation is the expectation kernel mean map regarding the density p and the second one is the empirical kernel mean map with the finite sample set $X = \{X_1, \ldots, X_M\}$. The expectation kernel mean map $\mu[p]$ is an element in the RKHS $\mathcal{H}$ as long as $\mathrm{E}_{x \sim p}[K(x,x)] < \infty$ [135]. Suppose that the samples from X are independently drawn from the same probability distribution with density p, then $\mu[p]$ can be approximated by $\mu[X]$ [136], with the bound of the deviation $\|\mu[p] - \mu[X]\|_{\mathcal{H}}$ with the probability given by [141]. The space of all probability distributions forms a convex set $\mathcal{P}$; thus, the image of the expectation kernel mean map $\mathcal{M} := \{\mu[p], \forall p \in \mathcal{P}\}$ is also convex and is called the marginal polytope [135]. In terms of the RKHS embedding, the goal of the density estimation is to find an optimal probability density $\hat{p} \in \mathcal{P}$ such that the deviation $\|\mu[X] - \mu[\hat{p}]\|_{\mathcal{H}}$ is minimized. In practice, the density estimator $\hat{p}$ is assumed as a mixture of a set of candidate densities or prototypes

p_i [135, 136] as

$$\hat{p} = \sum_{i=1}^{n} \alpha_i p_i, \tag{5.3}$$

where $\sum_i^n \alpha_i = 1$ and $\alpha_i \geq 0, \forall 1 \leq i \leq n$. Let us define the subset $\mathcal{P}_0$ of $\mathcal{P}$ as

$$\mathcal{P}_0 := \left\{ \hat{p} = \sum_{i}^{n} \alpha_i p_i | \sum_{i=1}^{n} \alpha_i = 1 \text{ and } \alpha_i \geq 0, \quad \forall 1 \leq i \leq n \right\}.$$

It can be seen that $\mathcal{P}_0$ is a convex hull of the prototypes since $\hat{p}$ is a convex combination of the candidate densities. The density estimation amounts to solving the minimization problem restricted to a convex set $\mathcal{P}_0$ as

$$\min_{\hat{p} \in \mathcal{P}_0} \left\| \mu\left[X\right] - \mu\left[\hat{p}\right] \right\|_{\mathcal{H}}^2. \tag{5.4}$$

Explicit expansion of Eq. (5.4) leads to solving a quadratic program for $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ as the following [136]

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \frac{1}{2} \boldsymbol{\alpha}^T \left(\mathbf{Q} + \lambda \mathbf{I}\right) \boldsymbol{\alpha} - \mathbf{q}^T \boldsymbol{\alpha} \\ &\text{s.t.} \sum_{i=1}^{n} \alpha_i = 1 \\ &\alpha_i \geq 0, \quad \forall 1 \leq i \leq n, \end{aligned} \tag{5.5}$$

where λ is a regularization constant to prevent overfitting, and $\mathbf{I}$ is the identity matrix. $\mathbf{Q} = \left[Q_{ij}\right]_{n \times n}$ is a matrix, and $\mathbf{q} = (q_1, \ldots, q_n)$ is a vector of length n, both of which are entries that depend on the kernel function. The matrix $\mathbf{Q}$ is positive definite; hence the above quadratic program (5.5) is a convex optimization problem.

5.2.2 High-order simulation method in spatial Legendre moment kernel space

5.2.2.1 SLM reproducing kernel

The motivation for applying statistical learning to the high-order simulation is to match the high-order spatial statistics of the output realizations to the training data through the learning process. This goal is achieved by the learning procedure in a newly defined kernel space, while the kernel is defined as

$$K\left(\boldsymbol{X}, \boldsymbol{Y}\right) = \prod_{i=0}^{N} \left[\sum_{w=0}^{W} \left(w + \frac{1}{2}\right) P_w\left(x_i\right) P_w\left(y_i\right) \right], \tag{5.6}$$

and is called a spatial Legendre moment kernel (SLM-kernel for short) of order W, where $\boldsymbol{X}, \boldsymbol{Y} \in [-1,1]^{N+1}, \boldsymbol{X} = (x_0, x_1, \ldots, x_N), \boldsymbol{Y} = (y_0, y_1, \ldots, y_N)$, and $P_w(\cdot)$ is the Legendre polynomial of order w defined on the interval $[-1,1]$.

As the name of the kernel suggests, one reason to define the SLM-kernel in the form of Eq. (5.6) is that past studies of high-order simulations based on Legendre-polynomial series have shown the capacity for capturing complex spatial patterns with spatial cumulants or spatial Legendre moments [17, 18, 133]. In other words, the SLM-kernel is constructed in a way that the distance between two distributions embedded into the kernel space actually represent the deviation of spatial Legendre moments from each other. The other reason stems from the fact that the computational model from Yao et al. [133] leads to a kernel-like expression of approximating the CPDF [cf. Equation Eq. (14) in [133]].

To prove that $K(\boldsymbol{X}, \boldsymbol{Y})$ is a positive definite, one can first define a simpler function $k(s,t) = P_w(s) P_w(t), \forall s, t \in [-1,1]$ and show that it is a positive definite. In fact,

$$\forall n \geq 1, \quad \forall a_i, a_j \in \mathbb{R}, \quad \forall t_i \in [-1,1], \quad 1 \leq i, j \leq n,$$

it is easy to see that

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j P_w(t_i) P_w(t_j) = \left[\sum_{i=1}^{n} a_i P_w(t_i)\right]^2 \geq 0.$$

Therefore, $k(s,t)$ is a positive definite. Now, we denote

$$\begin{aligned} K'(\boldsymbol{X}, \boldsymbol{Y}) &= \sum_{w=0}^{W} \left(w + \frac{1}{2}\right) P_w(x_i) P_w(y_i) \\ &= \sum_{w=0}^{W} \left(w + \frac{1}{2}\right) k(x_i, y_i). \end{aligned}$$

$K'(\boldsymbol{X}, \boldsymbol{Y})$ is a positive definite because the weighted sum of positive definite functions with non-negative coefficients is also positive definite. Finally, $K(\boldsymbol{X}, \boldsymbol{Y})$ can be written as $K(\boldsymbol{X}, \boldsymbol{Y}) = \prod_{i=0}^{N} K'(\boldsymbol{X}, \boldsymbol{Y})$. Given that the finite product of positive definite functions is also a positive definite [140], it is proven that the function $K(\boldsymbol{X}, \boldsymbol{Y})$ is positive definite, and thus, it defines a reproducing kernel.

5.2.2.2 Sequential simulation via statistical learning in SLM-kernel space

The implementation of a high-order stochastic simulation is under the framework of a sequential simulation [23]. By means of decomposing the multivariate probability distribution

into a consecutive set of univariate distributions, the simulation is carried out sequentially to generate random values from conditional distributions per a random path visiting the simulation grid. Specifically, let's let us denote the random field to be simulated as $\mathbf{Z}(\mathbf{u})$, which composes a multivariate distribution regarding the variable locations $\mathbf{u}$ at a discrete simulation grid. Suppose an arbitrary node Z_0 to be simulated within a random path is located at $\mathbf{u}_0$ with a neighborhood Λ of N conditioning data that contains either the sample data or the previously simulated nodes along the random path. Without loss of generality, the key problem in the stochastic simulation is to find an estimation o f the CPDF $f(Z_0|\Lambda)$, given the center node Z_0 and the N conditioning data. From the spatial configuration of the neighborhood, a spatial template can be extracted as $\mathbf{T} = (\mathbf{u}_0, \mathbf{u}_0 + h_1, \ldots, \mathbf{u}_0 + h_N)$, where $h_1, \ldots, h_N$ are distance vectors of the location of each conditioning data from the center node u_0. Clearly, statistical learning for the simulation aims to learn a target probability distribution from the available training data, and this turns out to be minimizing the distance of the empirical distribution and the target distribution after embedding them into the SLM-kernel space. By the definition of the Dirac delta function, one can define an empirical probability density function (EPDF) [115] corresponding to a sample set X of size M as

$$f_{\text{emp}}(x) = \frac{1}{M} \sum_{i=1}^{M} \delta(x - X_i). \tag{5.7}$$

Then, the empirical kernel mean map $\mu\,[X]$ can be rewritten as a convolution with the kernel K as

$$\mu_K[f_{\text{emp}}] \doteq \mu[X] = \int f_{\text{emp}}(x) K(x, \cdot) \mathrm{d}x. \tag{5.8}$$

Similarly, the expectation kernel mean map $\mu\,[p]$ can also be written as

$$\mu_K[p] = \int p(x) K(x, \cdot) \mathrm{d}x. \tag{5.9}$$

In this way, both the empirical kernel mean map $\mu[X]$ and the expectation kernel mean map $\mu[p]$ can be regarded as an integral operator μ_K determined by the kernel K acting on the EPDF or the PDF. The convolution of the density function with kernels can be analogous to the regularization of the integral operator to solve the ill-posed problem of density estimation [110, 142].

Given the above-mentioned template $\mathbf{T} = (\mathbf{u}_0, \mathbf{u}_0 + h_1, \ldots, \mathbf{u}_0 + h_N)$ and the replicate encountered in the TI as $\boldsymbol{\zeta}_t = (\zeta_{t,0}, \zeta_{t,1}, \ldots, \zeta_{t,N})$ corresponding to $\mathbf{T}$, the EPDF f_{emp} embedded

in the SLM-RKHS is identical to the density estimator in [133] in the kernel form as

$$\mu_K[f_{\text{emp}}] = \frac{1}{M}\sum_{t=1}^{M} K(\boldsymbol{\zeta}_t, \cdot). \tag{5.10}$$

Furthermore, under the sequential simulation framework, the CPDF $f(Z_0|\Lambda)$ can be mapped into a lower-dimensional kernel space through decomposition of the kernel space, so that the high-order simulation can be reduced to a one-dimensional optimization problem. Note that the kernel K in Eq. (5.6) can be decomposed as a product of lower-dimensional kernels K_0 and K_N as

$$K_0(x_0, y_0) = \sum_{w=0}^{W}(w + \frac{1}{2})P_w(x_0)P_w(y_0), \tag{5.11}$$

and

$$K_N(\boldsymbol{X'}, \boldsymbol{Y'}) = \prod_{i=1}^{N}[\sum_{w=0}^{W}(w + \frac{1}{2})P_w(x_i')P_w(y_i')], \tag{5.12}$$

where K_0 is one-dimensional and K_N is N-dimensional with $\boldsymbol{X'} = (x_1, \ldots, x_N), \boldsymbol{Y'} = (y_1, \ldots, y_N)$. Through marginalization of Eq.(5.10), the approximation of the CPDF $\tilde{f}_W(z_0|\Lambda)$ can be rewritten in terms of the kernels as

$$\tilde{f}_W(z_0|\Lambda) = \frac{\sum_{t=1}^{M} K_0(\zeta_{t,0}, z_0) \cdot K_N(\boldsymbol{\zeta}_t', \Lambda)}{\sum_{t=1}^{M} K_N(\boldsymbol{\zeta}_t', \Lambda)}, \tag{5.13}$$

where $\boldsymbol{\zeta}_t' = (\zeta_{t,1}, \ldots, \zeta_{t,N})$. By letting

$$\beta_t = \frac{K_N(\boldsymbol{\zeta}_t', \Lambda)}{\sum_{t=1}^{M} K_N(\boldsymbol{\zeta}_t', \Lambda)}, \tag{5.14}$$

the approximation of the CPDF $\tilde{f}_W(z_0|\Lambda)$ can be expressed as

$$\tilde{f}_W(z_0|\Lambda) = \sum_{t=1}^{M} \beta_t \cdot K_0(\zeta_{t,0}, z_0). \tag{5.15}$$

From Eq. (5.15), it turns out that the approximated CPDF $\tilde{f}_W(z_0|\Lambda)$ is a linear combination of kernel bases, and therefore, it lies in the SLM-RKHS with the kernel K_0. Furthermore, it can be regarded as the embedding of the empirical CPDF into the SLM-RKHS. In other words, the kernel mean map μ_{K_0} for the conditional distributions can be defined as

$$\mu_{K_0}[f_{\text{emp}}(z_0|\Lambda)] = \sum_{t=1}^{M} \beta_t \cdot K_0(\zeta_{t,0}, \cdot), \tag{5.16}$$

and

$$\mu_{K_0}[f(z_0|\Lambda)] = \int f(z_0|\Lambda)K_0(z_0,\cdot)\mathrm{d}z_0 = E[K_0(z_0,\cdot)], \tag{5.17}$$

where Eqs. (5.16) and Eq. (5.17) correspond to the SLM-RKHS embedding of the empirical CPDF and the target CPDF, respectively. Assuming that the CPDF can be expressed as the convex combination of some candidate distributions p_i as in Eq. (5.3), such that $f(z_0|\Lambda) \in \mathcal{P}_0$, then the density estimation for the CPDF can be solved by a similar minimization problem as Eq. (5.4) with the kernel mean map changing to μ_{K_0}. Explicit expansion of the minimization problem leads to a quadratic program similar to Eq. (5.5), whereas the matrix $\mathbf{Q}$ and the vector $\mathbf{q}$ are expressed as

$$Q_{ij} = E_{z_0\sim p_i, z_0'\sim p_j}[K_0(z_0, z_0')], \tag{5.18}$$

$$q_j = \sum_{t=1}^{M} \beta_t \cdot E_{z_0\sim p_j}[K_0(\zeta_{t,0}, z_0)]. \tag{5.19}$$

Therefore, combining Eqs. (5.5), (5.11), (5.18) and (5.19), the RKHS embedding of the CPDF leads to a quadratic program expressed by the one-dimensional kernel K_0. The solution to the optimization problem will give the weights α_i of each candidate distribution p_i, which leads to a target distribution matching to the high-order spatial statistics of the available data.

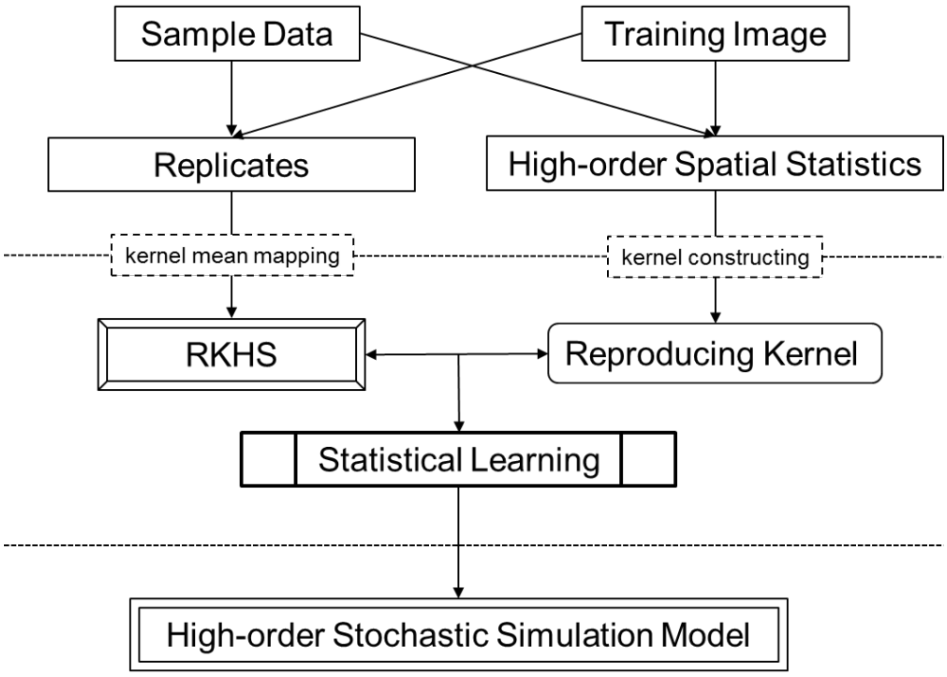


Figure 5.1 Workflow of high-order simulation via statistical learning

A general high-order stochastic simulation workflow via statistical learning is shown in Fig.

5.1. The main difference between the new high-order simulation workflow and the other geostatistical simulation methods is that the key element in the proposed workflow becomes the kernelization, including the kernel construction and the kernel mean mapping. A detailed implementation of the algorithm is given in Sect. 5.3.

5.3 Sequential Simulation Algorithm Based on Statistical Learning in SLM-RKHS

The SLM-RKHS embedding of the CPDF projects the density estimation in high-order stochastic simulation into a quadratic program in the feature space with SLM-kernel K_0 defined in the interval $[-1, 1]$. Hence, the sample data and the TI are first transformed into the interval $[-1, 1]$. The truncated normal densities on the interval $[-1, 1]$ are used as the prototypes. Let us denote the normal density with mean m_i and standard deviation σ as g_{σ,m_i} and its corresponding cumulative distribution function as G_{σ,m_i}. Then, the density functions of the prototypes are $p_i = g_{\sigma,m_i}/c_i$, with $c_i = G_{\sigma,m_i}(1) - G_{\sigma,m_i}(-1)$. Thus, the approximated CPDF can be expressed as

$$\hat{f}(z_0|\Lambda) = \sum_{i=1}^{n} \alpha_i g_{\sigma,m_i}(z_0)/c_i, \tag{5.20}$$

where n is the number of the prototypes. The computations of the matrix $\mathbf{Q}$ and the vector $\mathbf{q}$ are essential to build the quadratic program for solving the weights α_i. Further expansions of Q_{ij} and q_j in Eqs. (5.18) and Eq. (5.19) give

$$Q_{ij} = \sum_{w=0}^{W} (w + \frac{1}{2}) E_{z_0 \sim p_i}[P_w(z_0)] \cdot E_{z_0' \sim p_j}[P_w(z_0')], \tag{5.21}$$

$$q_j = \sum_{t=1}^{M} \beta_t \cdot (\sum_{w=0}^{W} (w + \frac{1}{2}) P_w(\zeta_{t,0}) E_{z_0 \sim p_j}[P_w(z_0)]). \tag{5.22}$$

As the computations of the coefficients β_t and the Legendre polynomial $P_w(\zeta_{t,0})$ are straightforward according to their definitions, the Legendre polynomial moment with the truncated normal density $E_{z_0 \sim p_i}[P_w(z_0)]$ remains the only term of more consideration. Here, a recursive algorithm to compute the Legendre polynomial moment $E_{z_0 \sim p_i}[P_w(z_0)]$ is developed. Let us denote $A_{w,i} = E_{z_0 \sim p_i}[P_w(z_0)]$ and $B_{w,i} = E_{z_0 \sim p_i}[z_0 P_w(z_0)]$. Note that $P_0(z_0) = 1$, and $P_1(z_0) = z_0, \forall z_0 \in [-1, 1]$. There are

$$A_{0,i} = 1, \tag{5.23}$$

and

$$A_{1,i} = B_{0,i} = m_i + \sigma^2[g_{\sigma,m_i}(-1) - g_{\sigma,m_i}(1)]/c_i. \tag{5.24}$$

The recursive relations of Legendre polynomials [93] are

$$(w+1)P_{w+1}(z_0) = (2w+1)z_0P_w(z_0) - wP_{w-1}(z_0), \tag{5.25}$$

and

$$(2w+1)P_w(z_0) = \frac{\mathrm{d}}{\mathrm{d}z_0}[P_{w+1}(z_0) - P_{w-1}(z_0)]. \tag{5.26}$$

By Eqs. (5.25) and (5.26) and through integration by parts, one can derive the following recursive equations

$$(w+1)A_{w+1,i} = (2w+1)B_{w,i} - wA_{w-1,i}, \tag{5.27}$$

and

$$\begin{aligned} B_{w,i} = {} & m_iA_{w,i} + \sigma^2[(-1)^w g_{\sigma,m_i}(-1) - g_{\sigma,m_i}(1)]/c_i \\ & + \sigma^2[(2(w-1)+1]A_{w-1,i} + \sigma^2[2(w-3)+1]A_{w-3,i} + \cdots . \end{aligned} \tag{5.28}$$

Combining with the initial conditions in Eqs. (5.23) and (5.24), Eqs. (5.27) and (5.28) form a complete recursive procedure to compute $E_{z_0\sim p_i}[P_w(z_0)]$. The computations in turn build the quadratic program for density estimation of the conditional probability distribution in the simulation.

In a situation with high-dimensional space, the location parameters m_i of the prototypes can be determined by clustering the available data. Here, since the density estimation problem is cast to the one-dimensional space by kernel decomposition, the locations of the prototypes are given by a set of peak points of the function from Eq.(5.15). Specifically, the interval $[-1, 1]$ is divided evenly into 100 subintervals and the prototypes are selected from the subintervals which contain the peak points of the function Eq. (5.15). This heuristic approach to selecting prototypes further simplifies the quadratic program and makes the simulation feasible for implementation. The scale parameter σ can be chosen by the method of stochastic gradient descent where the gradients can be derived from the recursive equations in Eq. (5.27) and (5.28).

In summary, the high-order stochastic simulation algorithm based on RKHS embedding (KERNELSIM hereafter for simplification) can be described as follows:

(1) Scale the property values of the samples and the TI to the interval [-1, 1].

(2) Generate a random path to visit the simulation grid.

(3) Pick one node from the random path to simulate, with the conditioning data taken from the neighborhood containing both the sample data and the previously simulated nodes.

(4) Replicates are scanned from the TI according to the template defined by the spatial configuration of the conditioning data.

(5) Compute the SLM-kernel moments to build the quadratic program.

(6) Solve the quadratic program to estimate of the CPDF with regard to the center node. Draw a random value from the CPDF as the data value of the center node.

(7) Repeat from step (3) until the simulation is completed.

(8) Back transform the property values of the simulation from [-1,1] to the original data space.

In a practical implementation, step (5) can be simplified to precompute the Legendre polynomial moments for each prototype distribution, as well as the Legendre polynomial values of the replicates, and therefore the computations can be greatly reduced at the cost of more memory usage. The solver for the quadratic program in step (6) applied to the present paper is based on the algorithm from [143].

The time complexity of the proposed algorithm is of polynomial time overall. Suppose that the size of the simulation grid is S and the size of the training data is M, the maximum order of the Legendre moments is W, the maximum number of conditioning data is N, and the number of the prototype distributions is n_p. Searching the replicates of the conditioning data from a regular grid takes $O(M \cdot N)$ operations. Computing the kernel moments and building the quadratic program takes $O\Big(M \cdot n_p(W^3 + W^2 N)\Big)$ arithmetic operations. Solving the quadratic program problem also takes polynomial time of $O(n_p^4 \cdot L)$, where L is the size of the problem encoding in binary [144]. Hence, the overall time complexity is a polynomial of $O\Big(S \cdot \Big(M \cdot n_p(W^3 + W^2 N) + n_p^4 \cdot L\Big)\Big)$.

5.4 Case Studies

5.4.1 Case study at a fully known reservoir

The porosity attributes from the Stanford V reservoir data set [129] are considered for simulation. Two horizontal sections at different depths are extracted from the reservoir, acting as the exhaustive image and the TI, respectively. For comparison, the two horizontal sections shown in Fig. 5.2 and Fig. 5.3 are selected to be the same ones used in a previous study

[133]. Firstly, the TI extracted from the original reservoir data set is rotated 45° clockwise to generate a new TI with seemingly different spatial structures, which are noted as TI-1 and TI-2 (Fig. 5.3 and Fig. 5.4), respectively. Furthermore, two different sets of sample data as DS-1 and DS-2 are drawn from the exhaustive image and are shown in Fig. 5.5 and Fig. 5.6, corresponding to the relatively sparse and dense samples, respectively. The main purposes of performing a simulation on these different cases are: (1) testing the sensitivity of KERNELSIM to the statistical conflicts between the sample data and the TI; (2) testing the impact of the number of sample data on the realization of KERNELSIM.

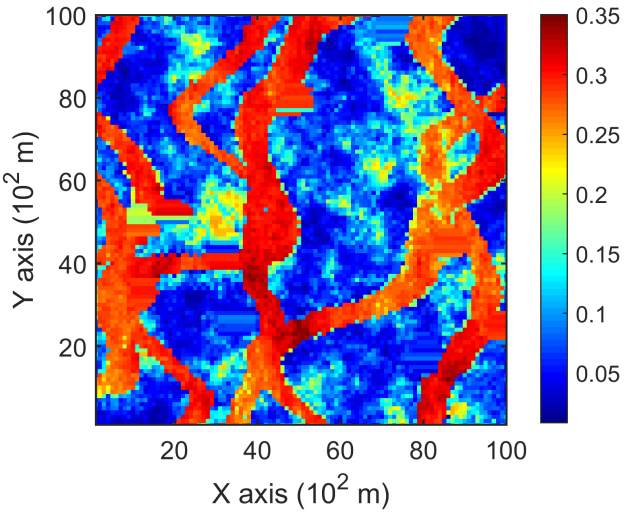


Figure 5.2 Exhaustive image: a horizontal section from a fully known reservoir

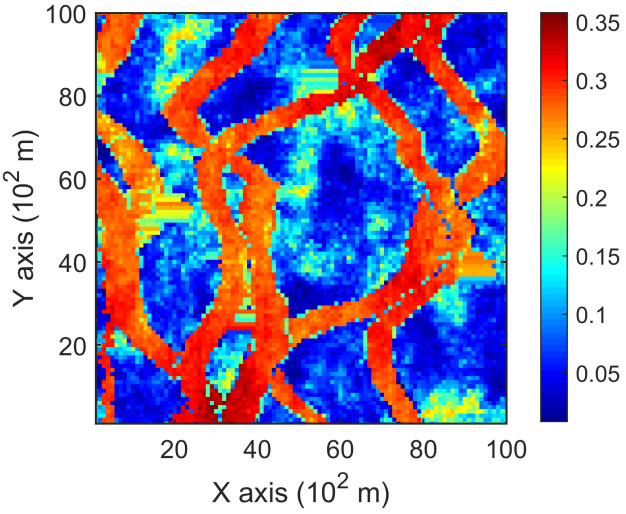


Figure 5.3 TI-1: another horizontal section from a fully known reservoir

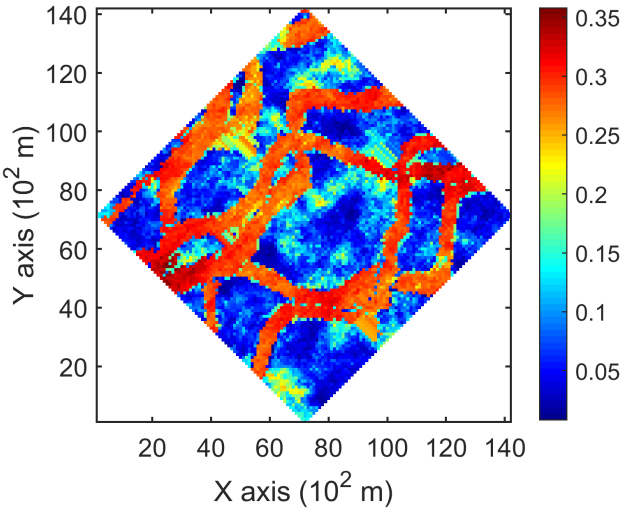


Figure 5.4 TI-2: rotation of TI-1 45° clockwise

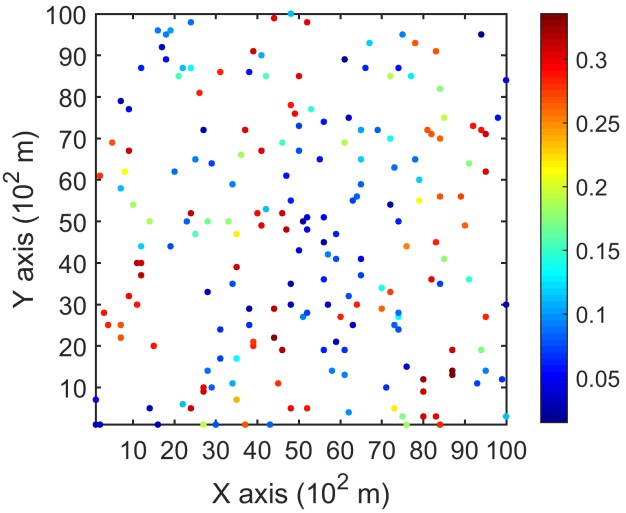


Figure 5.5 DS-1: data samples of 200 points drawn from the exhaustive image

5.4.1.1 Example 1

This example consists of simulation results generated by KERNELSIM with the TI-1 as the training image and DS-1 and DS-2 as the sample data sets. This example generally represents the situation where the sample data and the TI are of different origin but are sharing some similarity in spatial patterns. For instance, the channels in both the exhaustive image and TI-1 are preferential in the vertical directions.

Figure 5.7 shows one realization of KERNELSIM using TI-1 as the training image and with DS-1 and DS-2 as the sample data, respectively. For comparison, both realizations

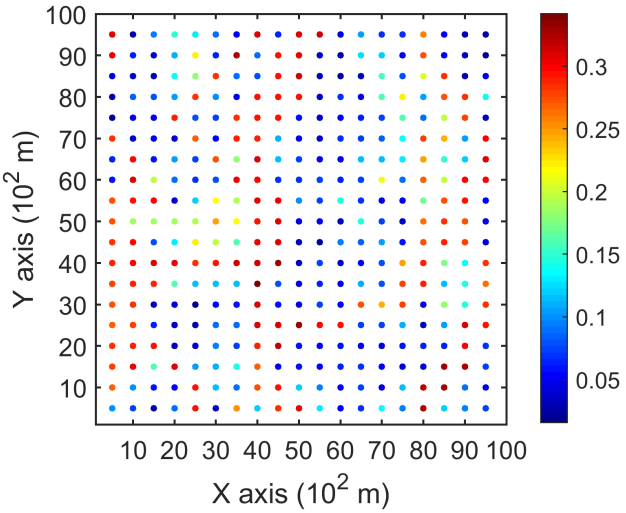
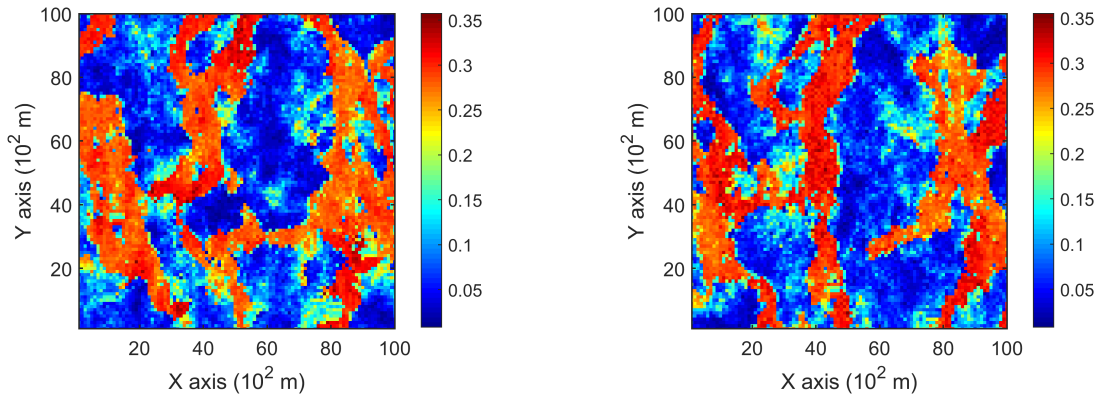


Figure 5.6 DS-2: data samples of 400 points drawn from the exhaustive image

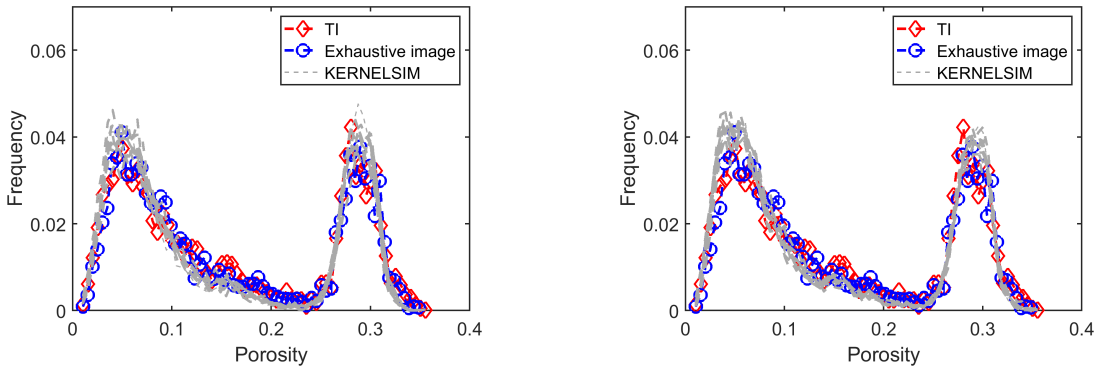
are generated by the same random path to visit the nodes on the grid. It is clear that both realizations reproduce the main spatial structures of the exhaustive image along the vertical channels from the visualization (Fig. 5.7). The realization shown in Fig. 5.7a is comparable to the case study in [133], and it shows that the present method reproduces channel connectivity better and eliminates the noisy points that appeared in the realizations generated using past approaches, which were caused by the impact of statistical conflicts between the sample data and the TI. Comparisons of the histograms and variograms of 10 realizations of KERNELSIM using either DS-1 or DS-2 as the sample data are illustrated in Fig. 5.8 and Fig. 5.9, respectively. The third-order cumulant maps of the sample sets DS-1 (smoothed for visualization) and DS-2 are shown in Fig. 5.10a, b. The cumulant maps of the exhaustive image and the TI are shown in Fig. 5.10c, d. For comparison, the third-order cumulant maps of the realizations of KERNELSIM using either the DS-1 or DS-2 as the sample data are shown in Fig. 5.10 e, f. Figure 5.10g, h shows the average third-order cumulant maps of 10 realizations using the DS-1 and DS-2 as the sample data, respectively. Similarly, a further comparison of fourth-order cumulant maps is displayed in Fig. 5.11. The spatial template for computing the fourth-order cumulant maps included directions along the X-axis, Y-axis and the diagonal direction. The fourth-order cumulant maps are scaled by their deviations for clearer visualization of the patterns. Both the third-order and the fourth-order cumulant maps clearly show that the KERNELSIM realization tends to have similar spatial patterns to the sample data and the exhaustive image. The above results show that the KERNELSIM method reproduces both the lower and higher spatial statistics of the underlying random field given that the TI and the sample data share some similarity

in their spatial distributions. Specifically, regardless the number of sample data being used, the main spatial features of the reservoir are retained in the realizations of KERNELSIM, as supported from the visual appearance of the vertical channels and the variograms, as well as from the cumulant maps.



(a) (b)

Figure 5.7 One realization from KERNELSIM using TI-1. a DS-1 as the sample data, b DS-2 as the sample data



(a) (b)

Figure 5.8 Histograms of 10 realizations of KERNELSIM using TI-1. a DS-1 as the sample data, b DS-2 as the sample data

5.4.1.2 Example 2

By rotating the TI-1 45° clockwise and creating a new training image as TI-2, shown in Fig. 5.4, there is seemingly a difference in the channel orientations between the TI-2 and the exhaustive image. Thus, this specific example aims to test the sensitivity of the KERNELSIM method to the more apparent statistical conflicts between the TI and the sample data. Fig. 5.12 shows one realization of KERNELSIM using TI-2 as the training image, along with

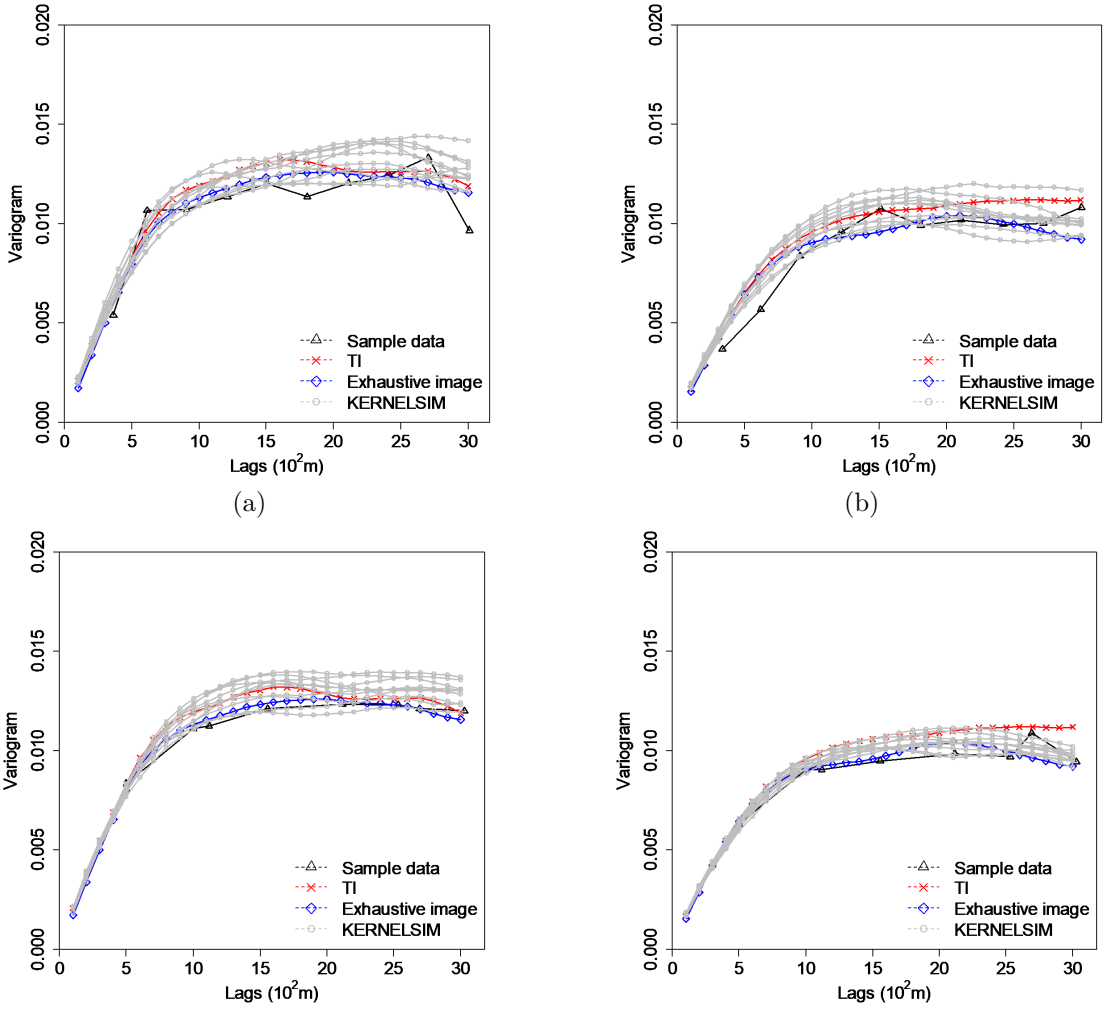


Figure 5.9 Variograms of 10 realizations of KERNELSIM using TI-1. a, b Along the X and Y axes with DS-1 as the sample data; c, d, along the X and Y axes with DS-2 as the sample data

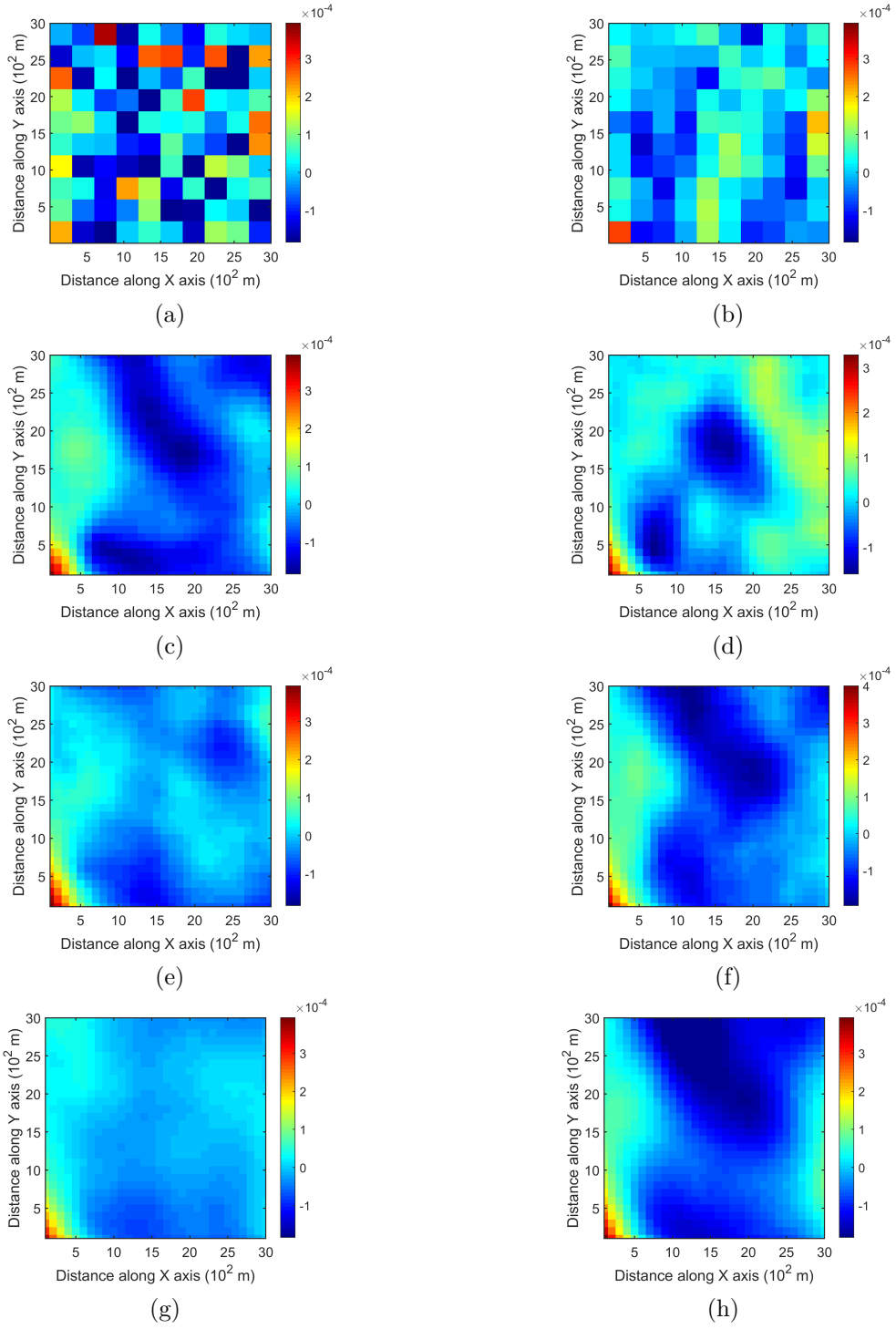


Figure 5.10 Third-order cumulant maps of a DS-1, b DS-2, c exhaustive image, d TI-1, e realization in Fig. 5.7a with DS-1 as the sample data, f realization in Fig. 5.7b with DS-2 as the sample data, g 10 realizations in average with DS-1 as the sample data, and h 10 realizations in average with DS-2 as the sample data

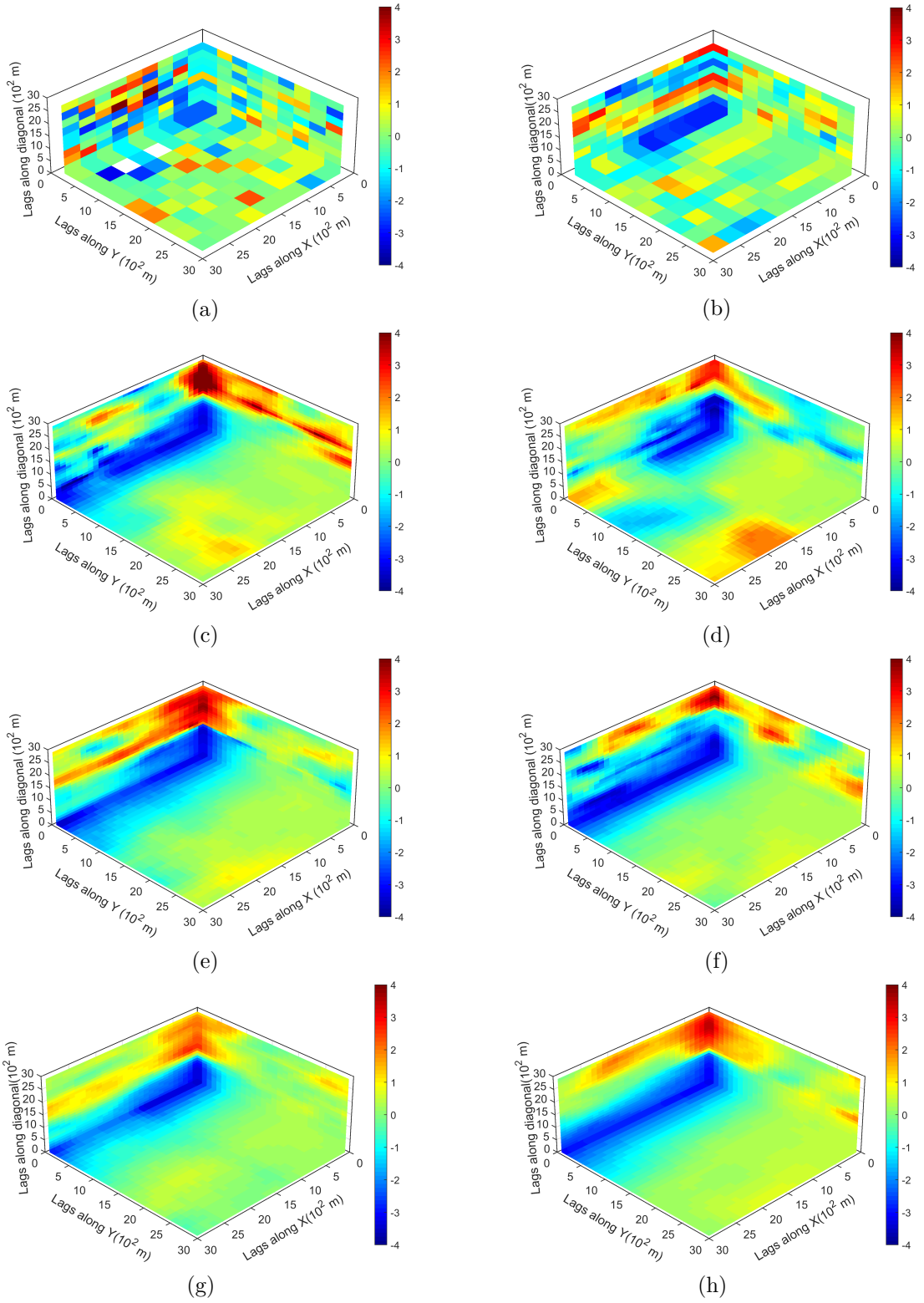


Figure 5.11 Fourth-order cumulant maps of a DS-1, b DS-2, c exhaustive image, d TI-1, e realization in Fig. 5.7a with DS-1 as the sample data, f realization in Fig. 5.7b with DS-2 as the sample data, g 10 realizations in average with DS-1 as the sample data, and h 10 realizations in average with DS-2 as the sample data

DS-1 and DS-2 as the sample data, respectively. Interestingly, even with relatively sparse sample data DS-1, the realization of KERNELSIM still reflects the vertical channels well. The same phenomena can also be observed in the realization using the denser sample data DS-2. Comparisons of the histograms and the variograms are shown in Figs. 5.13 and 5.14, respectively. Further, a comparison of high-order spatial statistics is shown in Fig. 5.15 and 5.16 in a similar way as in Example 1. While the third-order and the fourth-order cumulant maps of the TI and the exhaustive image are very different, the cumulant maps of the realizations still maintain the main spatial features of the one from the exhaustive image. This specific example shows that the KERNELSIM method is capable of generalizing the learning model to adapt to situations in the presence of statistical conflicts between the sample data and the TI. Of note, even with relatively sparse sample data, the proposed method can generate realizations with a reasonable reproduction of spatial statistics of the sample data from the lower to the higher orders.

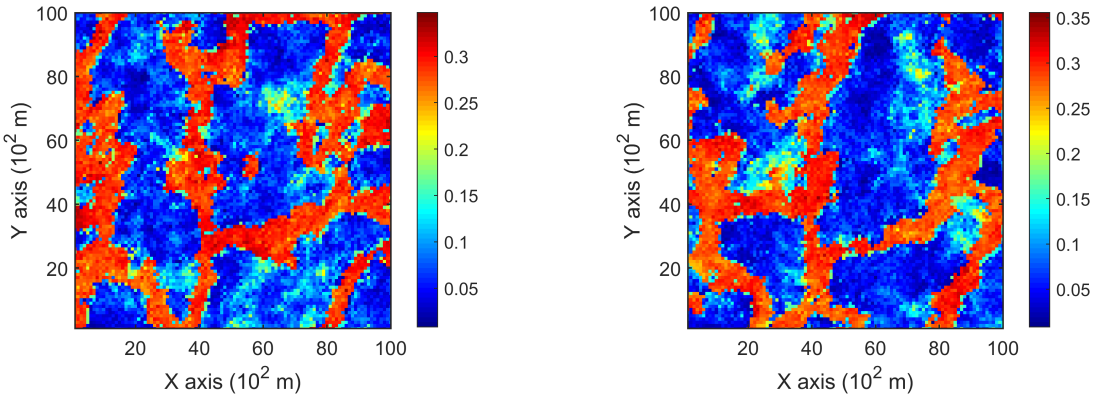


Figure 5.12 One realization from KERNELSIM using TI-2. a DS-1 as the sample data, b DS-2 as the sample data

5.4.1.3 Conditional probability on different spatial patterns

Three configurations of the conditioning data are intentionally picked at different locations to represent the typical spatial patterns that are possibly encountered in the data event. The KERNELSIM method is applied to generate the conditional probability distributions on these different spatial patterns to compare the behaviors of the CPDF at different locations (Fig. 5.17). Since the attribute values are transformed to the domain [-1, 1] of Legendre polynomials, both the conditioning data and the CPDFs are also in this domain. Fig. 5.17a shows the pattern of transition between lower values and higher values, which usually happens near the boundary of the channels in the exhaustive image, while Fig. 5.17b shows its

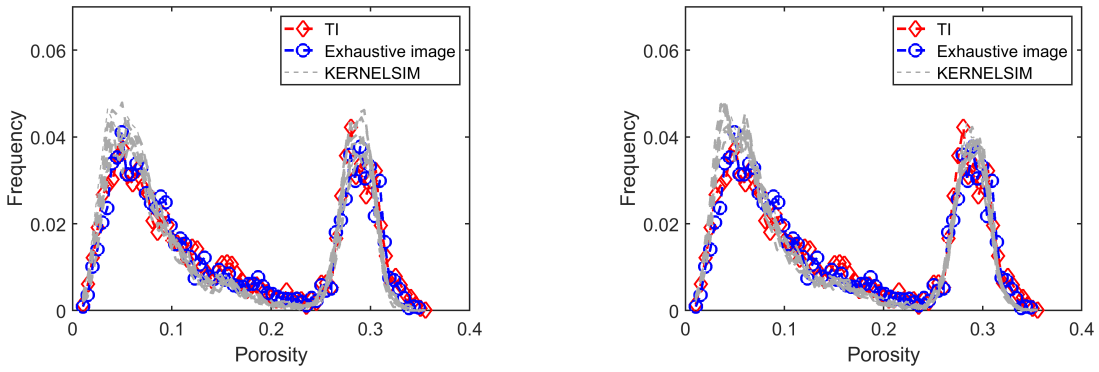


Figure 5.13 Histograms of 10 realizations of KERNELSIM using TI-2. a DS-1 as the sample data, b DS-2 as the sample data

corresponding CPDF at the center node. In this case, the CPDF has two different modes at the values of -0.41 and 0.74, which interestingly implies that the possible prediction could either be a lower value or a higher value, while the higher value has a higher likelihood. It turns out that the true value at this location after transformation is 0.745. However, it should be noted here that this double-modal behavior is reasonable near the boundary of transitioning between lower and higher values. This kind of probability distribution cannot be characterized by the second-order geostatistical simulation methods based on Gaussian assumption. Fig. 5.17c, d shows the simulation behavior at a location where the center node is surrounded by nodes with relatively lower values. Again, the CPDF also shows a bimodal shape due to the big variation of the spatial patterns. Fig. 5.17e, f shows the behavior of simulation at a location where the center node is surrounded by nodes with relatively higher values. The CPDF exhibits a unimodal distribution as the variation in the spatial pattern is small. Although the behaviors of CPDF could be case-dependent due to different spatial distributions of attributes of interest, these experiments show that the CPDFs generated by KERNELSIM are driven by the training data instead of a fixed covariance function, and thus can reflect the characteristics of different spatial patterns. In fact, several past studies have also shown the advantage of high-order simulation methods in reproducing the complex spatial patterns over the traditional second-order simulation methods, such as sequential Gaussian simulation [18, 128, 132].

5.4.2 Case study at a gold deposit

The case study at a gold deposit is presented here to demonstrate the practical aspects and the performance of KERNELSIM in its application to a real-life example. The sample data

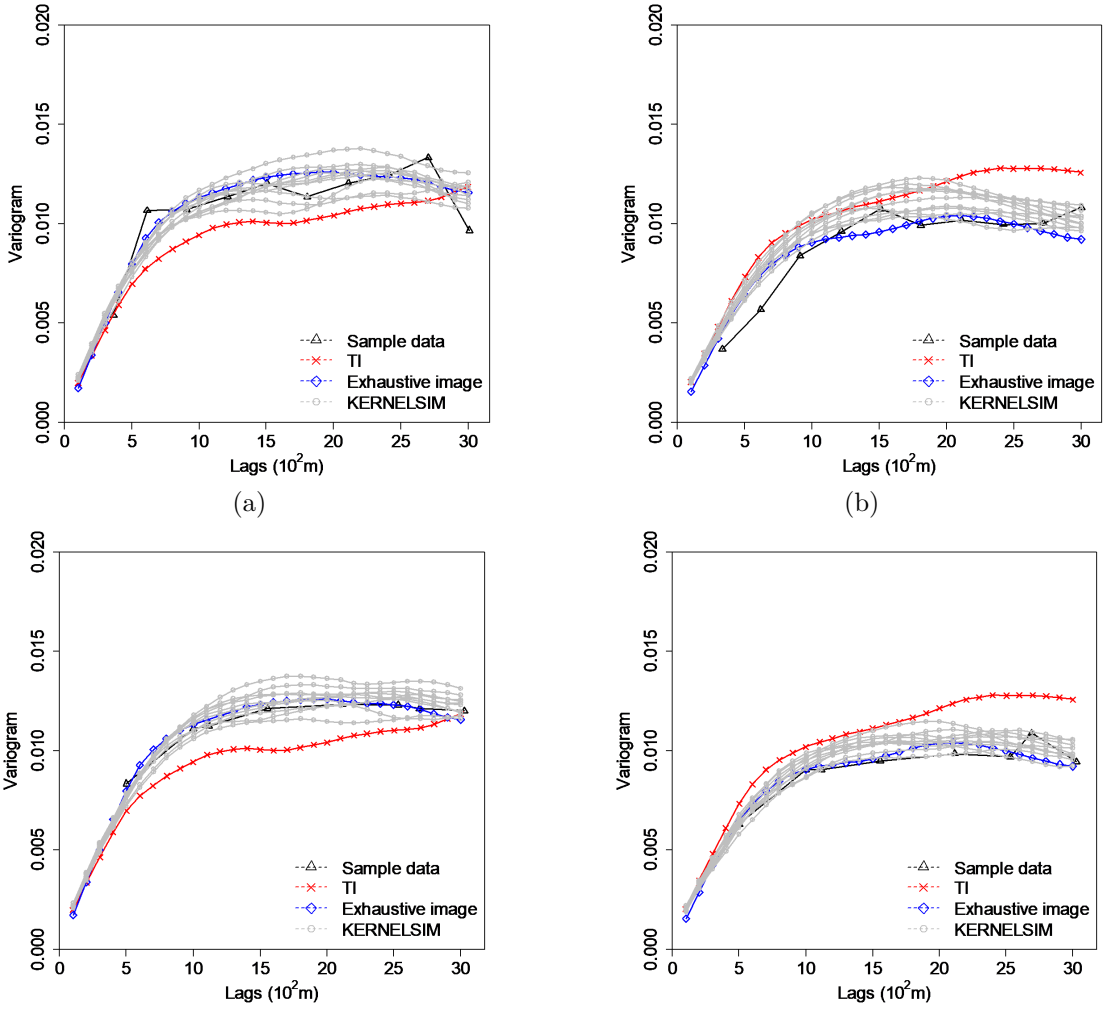


Figure 5.14 Variograms of 10 realizations of KERNELSIM using TI-2. a, b Along the X and Y axes with DS-1 as the sample data; c, d, along the X and Y axes with DS-2 as the sample data

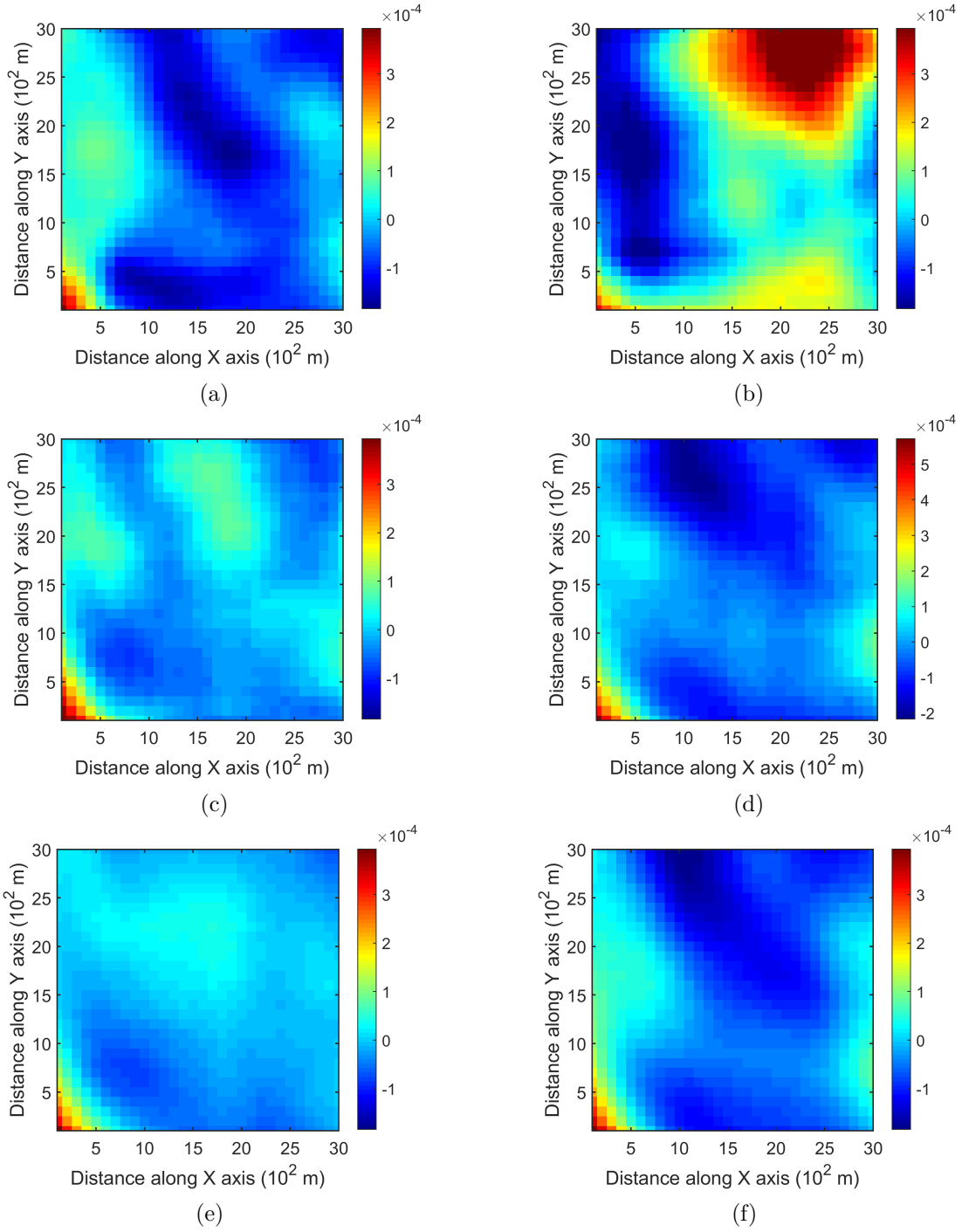


Figure 5.15 Third-order cumulant maps of a exhaustive image, b TI-2, c realization in Fig. 5.12a with DS-1 as the sample data, d realization in Fig. 5.12b with DS-2 as the sample data, e 10 realizations in average with DS-1 as the sample data, and f 10 realizations in average with DS-2 as the sample data

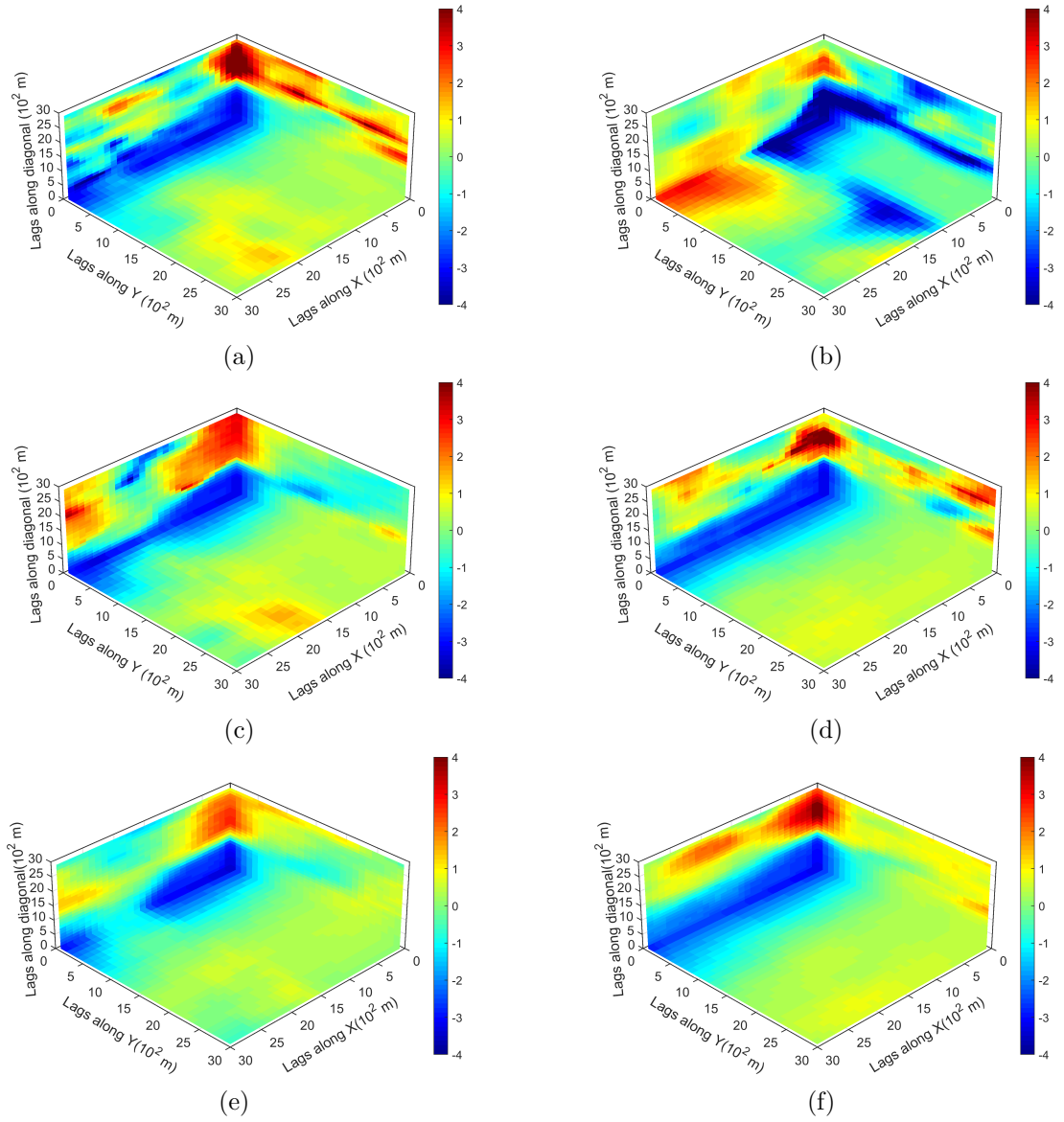


Figure 5.16 Fourth-order cumulant maps of a exhaustive image, b TI-2, c realization in Fig. 5.12a with DS-1 as the sample data, d realization in Fig. 5.12b with DS-2 as the sample data, e 10 realizations in average with DS-1 as the sample data, and f 10 realizations in average with DS-2 as the sample data

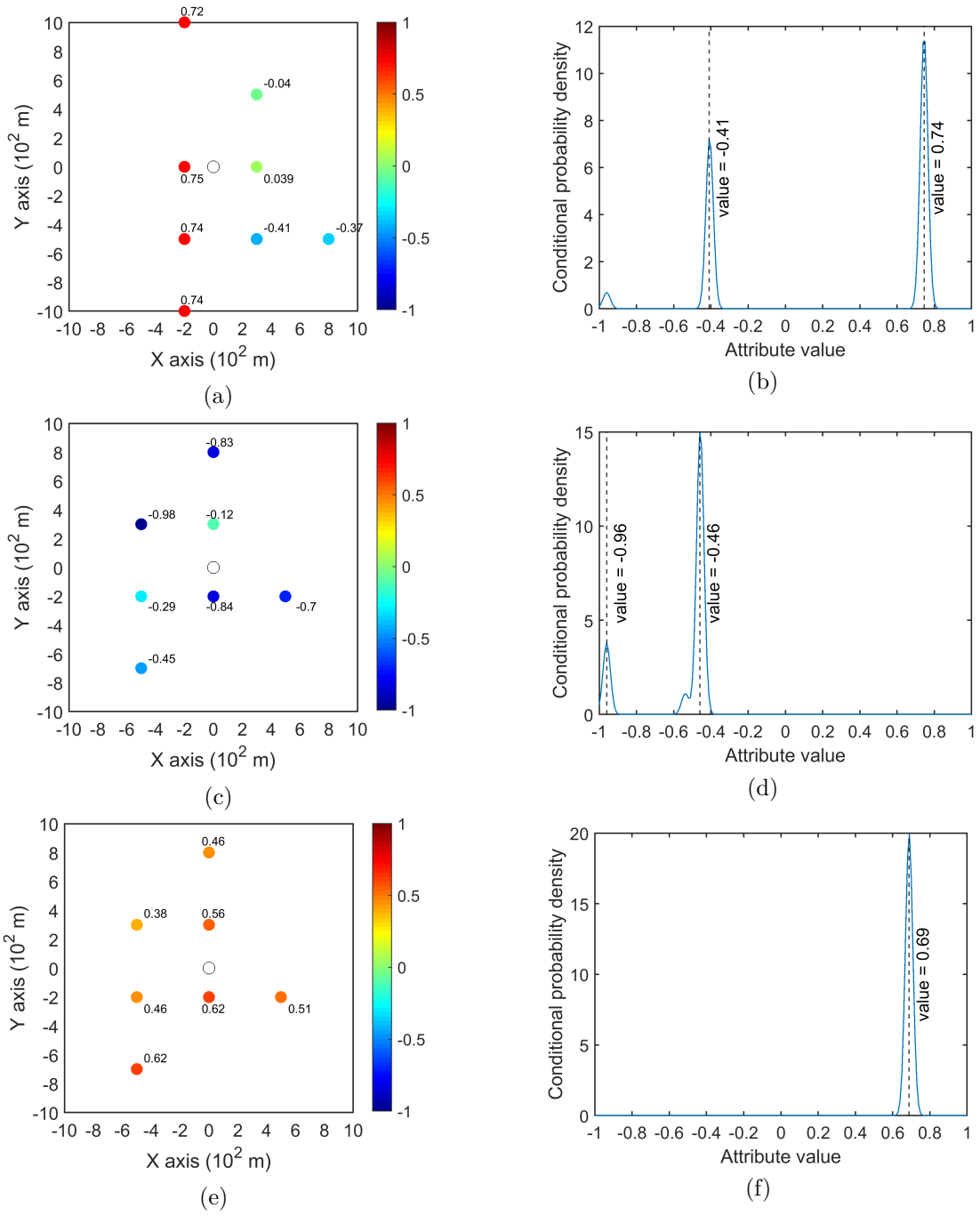


Figure 5.17 Behaviors of conditional probability distributions corresponding to conditioning data with different spatial patterns. The central circle represents the center node to be simulated, and the colored nodes are the conditioning data in the neighborhood

are from 407 exploration drill holes and are composited to 10 m in length. The TI comes from the blast hole data located at a mined-out area of the orebody. Fig. 5.18 shows the TI, a cross section of the TI and the sample data in a three-dimensional view. The TI is generated from the blast hole data assuming that the geological settings of the studied area are similar to the mined-out area, where conflicts would be mitigated by the statistical learning process dominated by the sample data. Fig. 5.19 shows cross sections of four different realizations of KERNELSIM for the gold deposit in a three-dimensional view. The histogram of the gold grades resembles the histogram of the sample data, as can be seen from Fig. 5.20. The variograms of the sample data and the TI are plotted for comparison with the variograms of 10 realizations of KERNELSIM from the gold deposit in Fig. 5.21. Figure 5.22 shows the third-order cumulant maps of the samples, the TI and the realization of KERNELSIM, respectively, along with the L-shape spatial template in the X-Y plane. Furthermore, the fourth-order cumulant maps of the samples, the TI and the realization of KERNELSIM are respectively displayed in Fig. 5.23. The results of the comparison in Figs. 5.22 and 5.23 show that the KERNELSIM reproduces the high-order spatial statistics of the sample data in addition to the lower-order statistics, even though the spatial patterns of the third-order and fourth-order cumulant maps of the TI are different to those of the sample data.

5.5 Conclusions

The paper presents a new high-order stochastic simulation framework based on statistical learning. Within this statistical learning workflow, the density estimation in the sequential simulation is kernelized, which renders it equivalent to solving a quadratic programming problem. The kernelization is approached by embedding the original data space into a kernel Hilbert space. A spatial Legendre moment reproducing kernel is proposed to construct an RKHS that can incorporate the high-order spatial statistics of the original data. In addition, a kernel decomposition technique is proposed to project the kernelization into a one-dimensional kernel Hilbert space to approach the sequential simulation procedure and to reduce computational complexity. The proposed statistical learning framework is general and can cope with the possible statistical conflicts between the sample data and the TI. The implementation of the method presented, termed KERNELSIM, is tested in different case studies. The examples, which use a fully known reservoir, show that KERNELSIM can reproduce the main spatial patterns of the sample data. Notably, the generalization capacity of the proposed method mitigates the statistical conflicts between the sample data and the TI and retains high-order statistical features from the sample data. The two examples in the first case study also provide some insights on how the number of the sample data

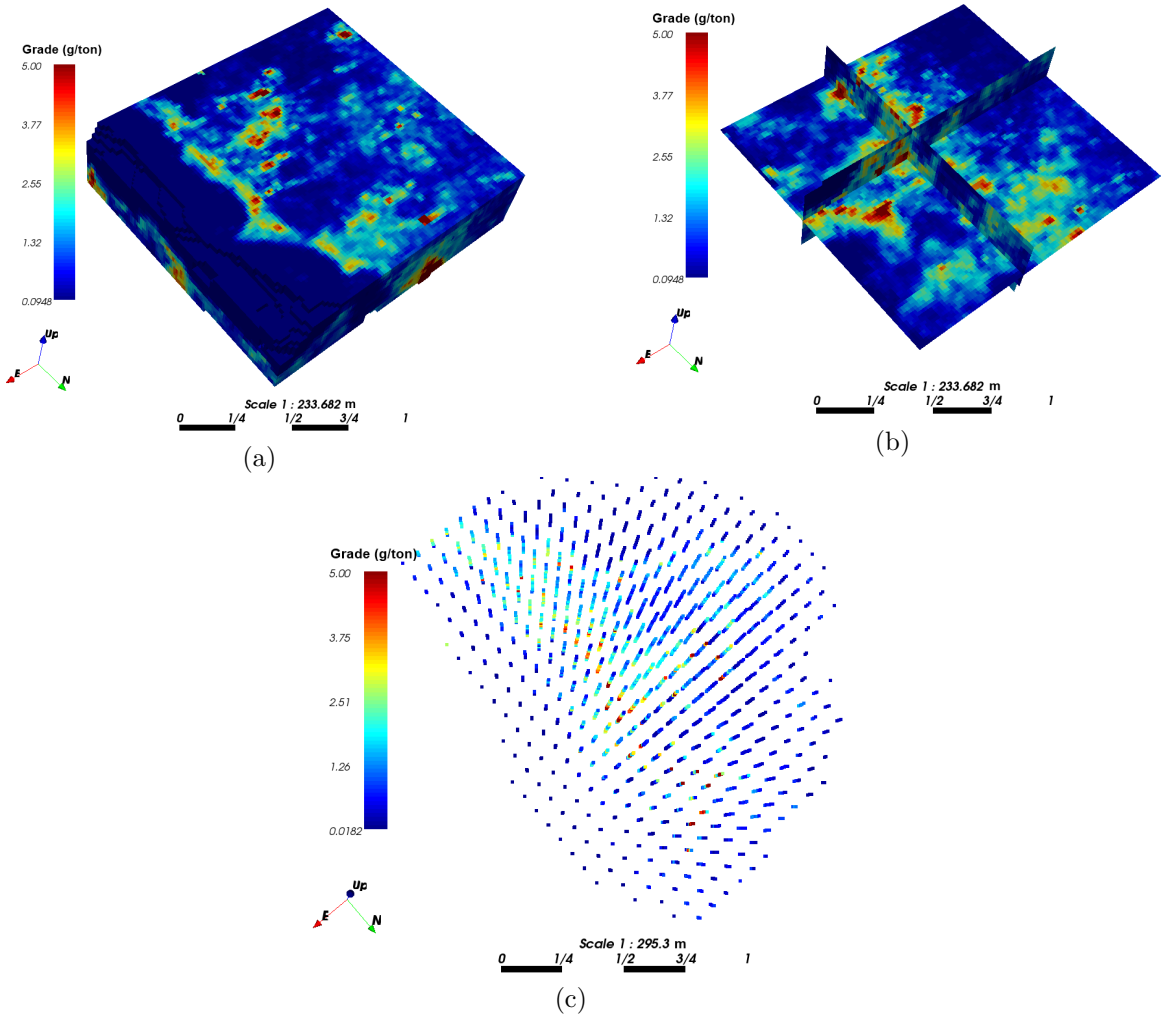


Figure 5.18 a TI, b a cross section of the TI, and c the sample data of the Au grades

and the relation of the sample data to the TI affect the simulation results. It should be noted that the simulation results only use the replicates from the TI to infer a conditional probability distribution. Hence, the proposed statistical framework provides an approach to condition the local probabilistic models learning from the TI to the existing configuration of the sample data based on the generalization capacity of the learning framework. However, the assumption made is that the TI shares some similarities in the local spatial structures with the sample data, even though their global structures could be different. The impact of the TI can also be reduced by only using replicates from the sample data, if the number of the replicates reaches a certain threshold of statistical significance, similarly to the approach adopted in previous publications [18, 133]. A case study at a gold deposit demonstrates the performance of KERNELSIM in a three-dimensional example. The results show that the KERNELSIM method reproduces the high-order spatial statistics of the drill hole samples

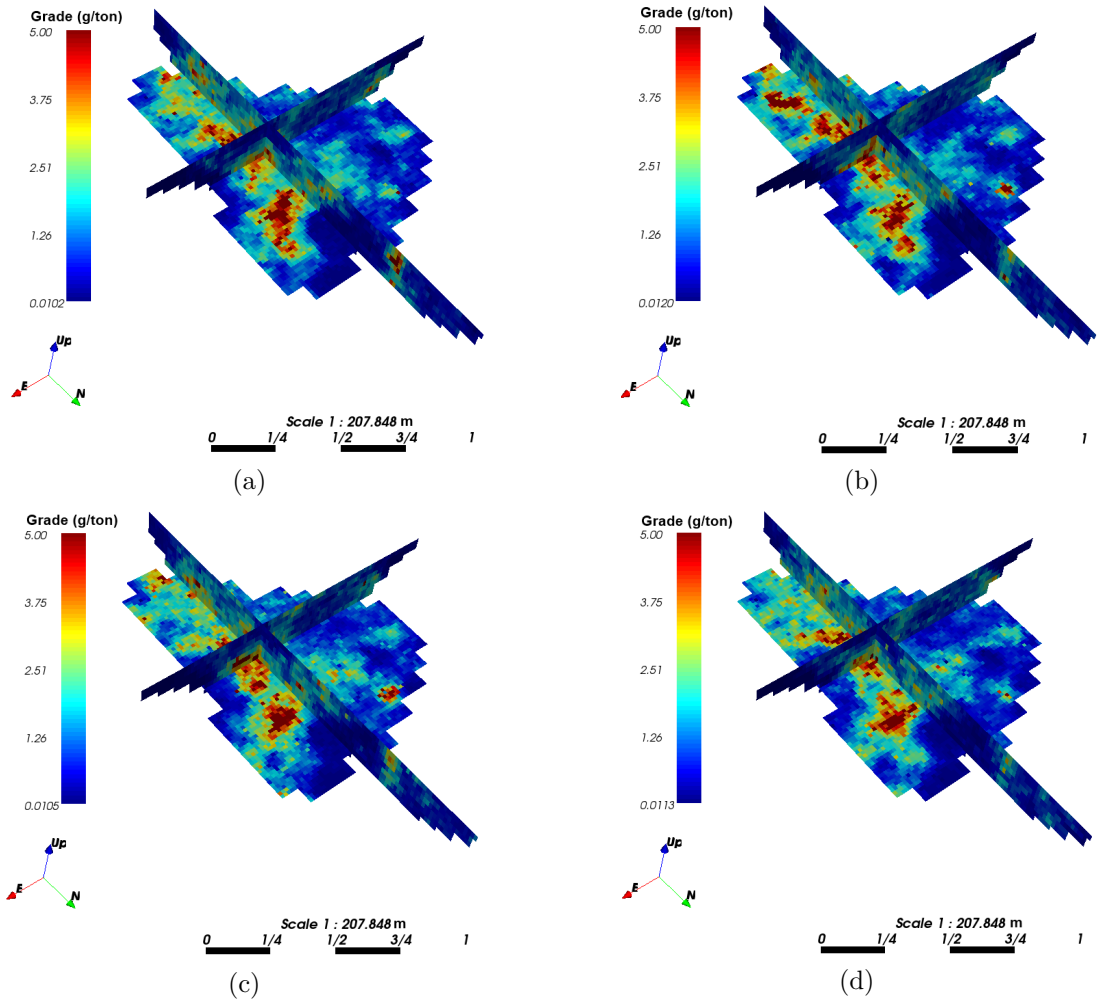


Figure 5.19 Cross sections of four different realizations of KERNELSIM of the Au grades

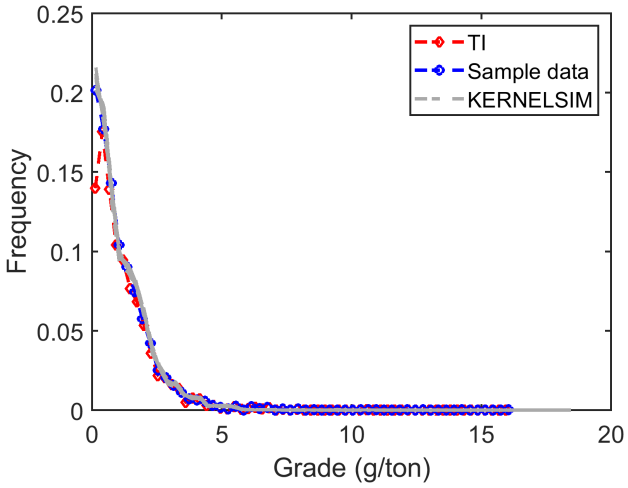


Figure 5.20 Histograms of 10 realizations of KERNELSIM for the Au grades of the gold deposit in comparison to the TI and the samples

well. Thus, the method provides an effective approach to simulate the orebody using the drill hole samples with the TI originating from a suitable mined-out part of the same deposit.

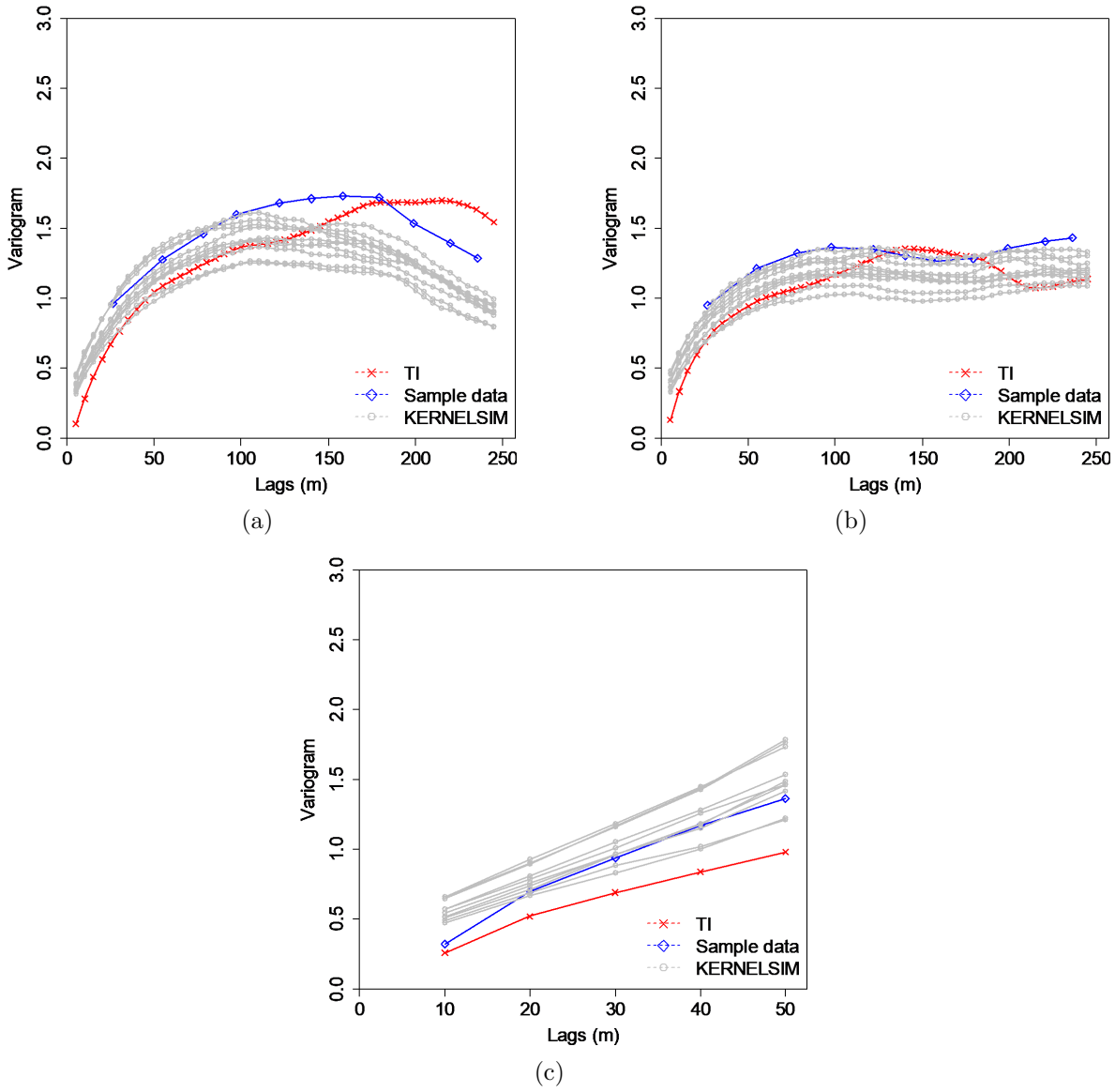


Figure 5.21 Variograms of 10 realizations of KERNELSIM for Au grades at the gold deposit along a E–W, b N–S, and c down drill holes, in comparison to the sample data and the TI

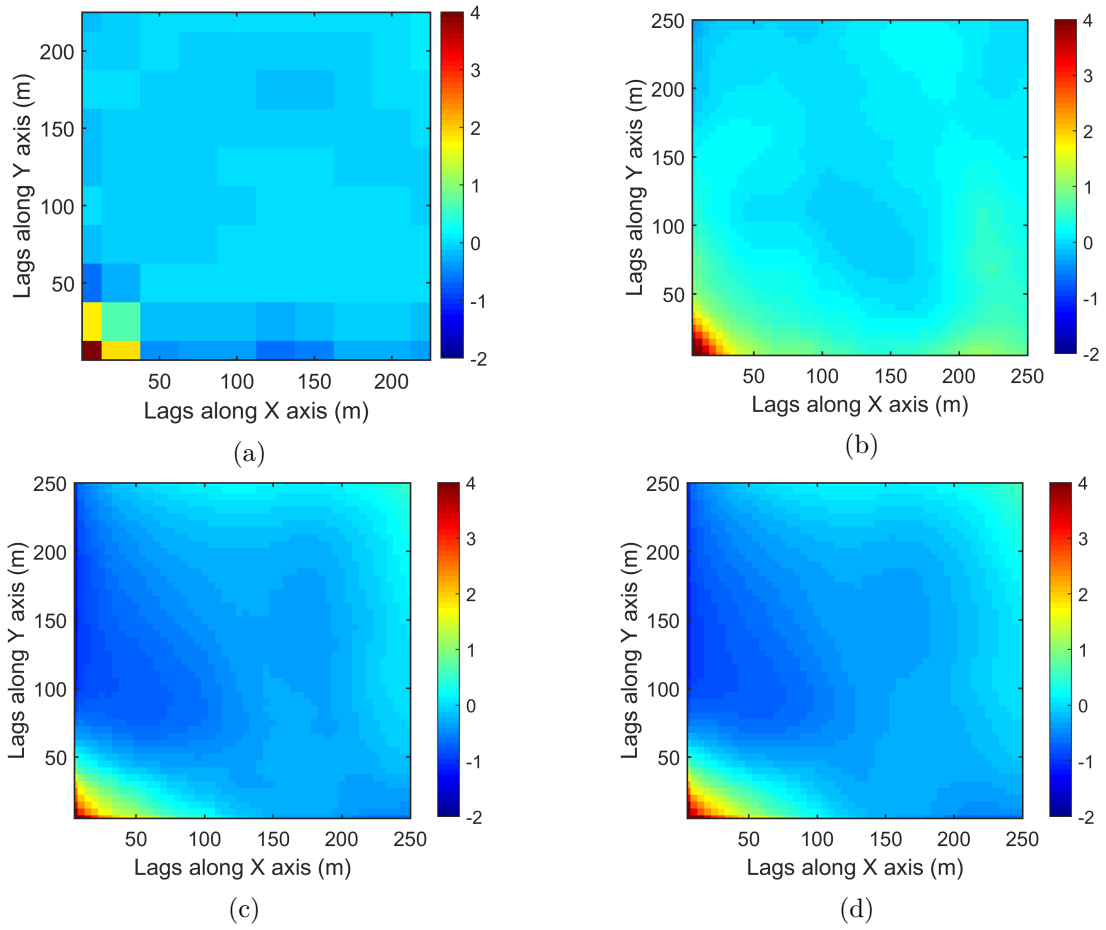


Figure 5.22 Third-order cumulant maps of a the sample data, b the TI, c the realization of KERNELSIM and d the 10 realizations of KERNELSIM in average

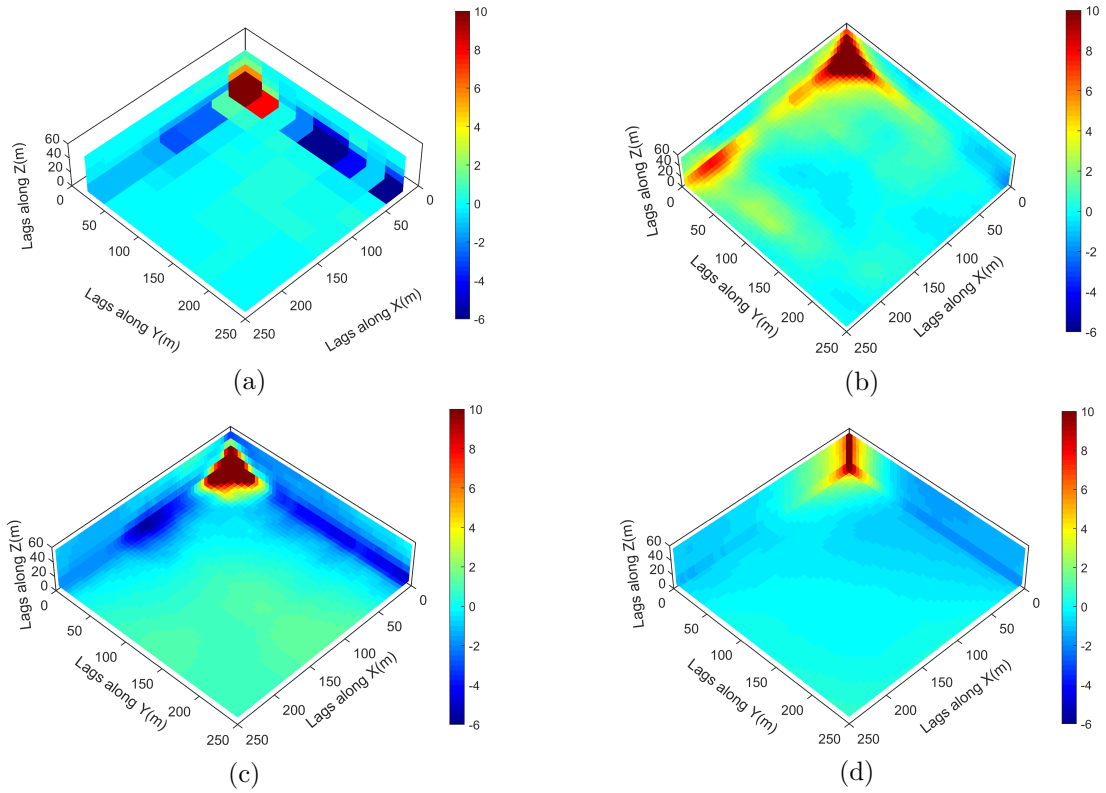


Figure 5.23 Fourth-order cumulant maps of a the sample data, b the TI, c the realization of KERNELSIM and d the 10 realizations of KERNELSIM in average

CHAPTER 6 ARTICLE 3: TRAINING-IMAGE FREE HIGH-ORDER STOCHASTIC SIMULATION BASED ON AGGREGATED KERNEL STATISTICS

Abstract: A training-image free, high-order sequential simulation method is proposed herein, which is based on the efficient inference of high-order spatial statistics from the available sample data. A statistical learning framework in kernel space is adopted to develop the proposed simulation method. Specifically, a new concept of aggregated kernel statistics is proposed to enable sparse data learning. The conditioning data in the proposed high-order sequential simulation method appear as data events corresponding to the attribute values associated with the so-called spatial templates of various geometric configurations. The replicates of the data events act as the training data in the learning framework for inference of the conditional probability distribution and generating simulated values. These replicates are mapped into spatial Legendre moment kernel spaces and the kernel statistics are computed thereafter, encapsulating the high-order spatial statistics from the available data. To utilize the incomplete information from the replicates, which partially match the spatial template of a given data event, the aggregated kernel statistics combine the ensemble of the elements in different kernel subspaces for statistical inference, embedding the high-order spatial statistics of the replicates associated with various spatial templates into the same kernel subspace. The aggregated kernel statistics are incorporated into a learning algorithm to obtain the target probability distribution in the underlying random field, while preserving in the simulations the high-order spatial statistics from the available data. The proposed method is tested using a synthetic data set, showing the reproduction of the high-order spatial statistics of the sample data. The comparison with the corresponding high-order simulation method using TIs emphasizes the generalization capacity of the proposed method for sparse data learning.

Keywords: High-order sequential simulation; Statistical learning; Spatial statistics; Kernel space

6.1 Introduction

Stochastic simulation methods are used to quantify the uncertainty of spatially distributed attributes of geological and other natural phenomena. It is well known that the conventional

Submitted: Yao L, Dimitrakopoulos R, Gamache M (2020) Training-image-free high-order stochastic simulation based on aggregated kernel statistics. Mathematical Geosciences (Submitted)

second-order stochastic simulation methods are limited in reproducing the complex patterns or nonlinear features exhibited in the spatial attributes of interest [14, 145, 146]. The so-called multiple point simulation (MPS) methods [12, 56, 57, 60, 65, 66, 84, 120, 122, 147] have been developed to address the limitation of conventional simulation methods based on the concept of multiple point statistics. The multiple point simulation framework introduced training images (TI) as statistical analogs of the spatial attributes under consideration. The multiple point statistics are either (a) captured by occurrences of data events formed by indicators at multiple locations inside the so-called spatial templates when the spatial attributes are categorical, or (b) generalized to continuous data as the pattern similarity among patches from the TI and the proceeding simulation. The multiple point statistics described in the MPS methods are based on a certain spatial template, however, are limited given that they do not consistently consider the lower-order spatial statistics in the related sub-templates. In addition, although the utilization of a TI as prior information to account for multi-point interactions of spatial attributes is conceptually appealing and justified [84], generally, the information from TI is not conditioned to the available data. Thus, the potential statistical conflicts existing between the sample data and the TI is a hinderance for the TI-driven MPS methods to reproduce the spatial patterns properly. This issue seems more prominent when the sample data are relatively dense, as in mining applications [126].

The high-order simulation methods provide a new framework to simulate complex spatial patterns, addressing the drawbacks in MPS methods as discussed in the related publications [17–19, 98, 99, 127, 132, 133, 148]. The high-order simulation methods equip the multiple-point spatial structures with well-defined mathematical entities, such as spatial cumulants or high-order spatial moments [17, 18, 128]. The random field model in the high-order simulation framework makes no assumption on any specific probability distribution. Instead, a Legendre polynomial expansion series is adopted to approximate the underlying distribution, where spatial cumulants are quantified to infer the expansion coefficients [18, 19]. To cope with the statistical conflicts between the samples and the TI, the high-order simulation methods take into account both the high-order spatial statistics from the sample data and the TI. However, the latter ones are only incorporated when the replicates from the sample data are insufficient for inference and, therefore, limit the influence of the TI on the realizations [18, 19]. Minniakhmetov and Dimitrakopoulos [98] propose a high-order simulation method without TI, which uses instead special relations of high-order indicator moments in boundary conditions related to a certain spatial template. However, these mathematical relations can only be established for categorical random variables. Yao et al. [148] propose a statistical learning framework of high-order simulation in kernel space by constructing a so-called spatial Legendre moment kernel from a new computational model of high-order simulation based

on spatial Legendre moments [133]. The proposed statistical learning framework in Yao et al. [148] demonstrates the advantage of its generalization capacity with regards to improving of the numerical stability, as compared to the previous high-order simulation methods. This generalization capacity also mitigates the statistical conflicts between the samples and the TI. This is due to the fact that the high-order spatial statistics are adjusted to the target probability distribution through the learning process, as opposed to directly being incorporated into the coefficients of polynomial expansion series as with the other methods. The simulation under a statistical learning framework [148] proceeds sequentially according to a random path based on the sequential decomposition of the multivariate distribution of the random field model [23, 25]. Specifically, the replicates are mapped onto the spatial Legendre moment space and the empirical kernel statistics are computed thereafter. The target probability distributions are also embedded into the same kernel space to obtain the expected kernel statistics. Matching these two elements in the kernel space leads to a minimization problem in the quadratic form determined by the kernel function. Solving the minimization problem leads to target probability distributions that comply with the high-order spatial statistics of the available data.

The present paper proposes fundamental adjustments of the above statistical learning framework so that it becomes more suitable for sparse data learning, thus allowing the development of a TI free high-order simulation method for the continuous spatial attributes. Since retrieving replicates that fully match the spatial template of the data events is difficult due to the sparsity of the sample data, it is worth noting that replicates that are partially matched to the spatial template may exist. These partially matched replicates, nevertheless, provide useful and relevant information to the related statistical inference, while determining how to utilize this incomplete information remains a challenge. The above-mentioned matters are addressed herein by a proposed concept of aggregated kernel statistics. More specifically, each spatial template is associated with a certain kernel subspace, such that any replicate associated with the same spatial template can be mapped onto an element of the corresponding kernel space. Accordingly, these mapped elements in the kernel subspaces are utilized to compute the kernel statistics. The kernel statistics in a set of kernel subspaces are combined to determine the aggregated kernel statistics through the relations introduced in this paper. Eventually, the aggregated kernel statistics are embedded into the kernel subspace corresponding to the conditional probability distribution encountered in the high-order sequential simulation framework, and the statistical learning algorithm is applied to approximate a conditional probability distribution.

The remainder of the paper is organized as follows. Firstly, the mathematical concepts and the proposed method are presented. Next, a case study from a synthetic data set is used

to assess the performance of the proposed method and demonstrate its practical aspects. Conclusions follow.

6.2 Method

Consider the spatial attributes of interest distributed on a discrete grid as a random field model denoted by $\mathbf{Z}(\mathbf{u})$ with $\mathbf{u} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n\}$ corresponding to various locations within the grid, then $\mathbf{Z}(\mathbf{u}) = \{Z(\mathbf{u}_1), Z(\mathbf{u}_2), \ldots, Z(\mathbf{u}_n)\}$ comprises a multivariate probability distribution $f_{\mathbf{Z}}$ given that $\mathbf{Z}(\mathbf{u}_i)$ representing random variables at location $\mathbf{u}_i (i = 1, \ldots, n)$. Under the sequential simulation framework [23], the joint probability distribution $f_{\mathbf{Z}}$ is decomposed into a sequence of conditional probability distributions following a random path to visit the entire simulation grid, random values are drawn from these conditional probability distributions sequentially along the random path to generate one realization. Both the available sample data and the previous simulated attribute values are considered as the conditioning data throughout the simulation process.

Without loss of generality, suppose that the current attribute $\mathbf{Z}(\mathbf{u}_0)$ to be simulated locates at $\mathbf{u}_0$, and the informed data $\{\zeta_1, \ldots, \zeta_N\}$ at the surrounding locations $\mathbf{u}_0 + \mathbf{h}_1, \ldots, \mathbf{u}_0 + \mathbf{h}_N$, consist of a data event as the conditioning data. From the geometric configuration of the data event, a spatial template $\mathbf{T} = \{\mathbf{u}_0, \mathbf{u}_0 + \mathbf{h}_1, \ldots, \mathbf{u}_0 + \mathbf{h}_N\}$ can be determined with the distance vectors $\mathbf{h}_1, \ldots, \mathbf{h}_N$ pointing outwards from the center $\mathbf{u}_0$ to the surrounding locations. Let the conditional probability density function (CPDF) be denoted as $f(z_0|\zeta_1, \ldots, \zeta_N)$, the key task to derive the CPDF is achieved by a statistical learning algorithm in kernel space herein. The related replicates associated with template $\mathbf{T}$ are retrieved from the sample data and these replicates are used as the training data of statistical learning to infer the underlying probability distribution. Specifically, the retrieved replicates are mapped to elements in kernel spaces to build kernel statistics carrying the high-order spatial information from the replicates. The aggregated kernel statistics are proposed allowing to incorporate the high-order spatial statistics from the ensemble of replicates with different spatial configurations. The target CPDF is then achieved by the statistical learning algorithm approaching the aggregated kernel statistics from the sample data.

6.3 Aggregation of Spatial Legendre Kernel Subspaces and Kernel Statistics

6.3.1 Spatial Legendre moment kernel subspaces

The kernel space is a Hilbert space defined through a positive kernel function. The spatial Legendre moment reproducing kernel (SLM-kernel) [148] is presented herein to carry the information of high-order spatial statistics so that the density estimation in the high-order sequential simulation could be achieved by a statistical learning process in kernel space. The SLM-kernel can be defined to associate a kernel subspace to random variables within a certain spatial template. Given a set of random variables $V = \{Z_0, Z_1, \ldots, Z_N\}$ with nodes corresponding to spatial template $\mathbf{T} = \{\mathbf{u}_0, \mathbf{u}_0 + \mathbf{h}_1, \ldots, \mathbf{u}_0 + \mathbf{h}_N\}$. The kernel subspace can be determined by a spatial Legendre moment reproducing kernel (SLM-kernel) as

$$K_V(\mathbf{X}, \mathbf{Y}) = \prod_{i=0}^{N} \Big[\sum_{w=0}^{W} (w + \frac{1}{2}) P_w(x_i) P_w(y_i) \Big], \tag{6.1}$$

where N corresponds to size of the spatial template, $\mathbf{X} = (x_0, x_1, \ldots, x_N)$, $\mathbf{Y} = (y_0, y_1, \ldots, y_N)$, and $P_w(\cdot)$ is the Legendre polynomial of order w defined on the interval $[-1, 1]$ and W is the maximal order of Legendre polynomials under consideration. Let the original data space denote as $\mathbb{E}$ and the kernel space associated to kernel K denote as $\mathcal{H}$, the canonical feature map [139], $\phi(t) : \mathbb{E} \to \mathcal{H}, t \mapsto K(\cdot, t), \forall t \in \mathbb{E}$, defines a valid feature map which takes an element from the original data space to an element in the kernel subspace. In other words, after the feature mapping, each element in the original data space $\mathbb{E}$ has a "representer" in the kernel space $\mathcal{H}$.

6.3.1.1 Aggregated SLM-Kernel Statistics

If a training image (TI) is provided as an exhaustive data set, most of the replicates of a data event fully match the spatial configuration of the data event while the partially matched ones are negligible. The replicates of a data event from the sample data, however, include both fully matched and partially matched replicates which correspond to different configuration of spatial templates. Therefore, the replicates are respectively mapped to different kernel subspaces. Kernel statistics, in general, means either the empirical statistics from the mapped elements or the expected statistics in the kernel subspaces, such as empirical mean and expectation. Equation (6.1) suggests that replicates associated with different spatial templates would be mapped to kernel subspaces with different kernel functions. The kernel statistics associated with different spatial templates, thus come from different subspaces and need to be combined appropriately to get the aggregated kernel statistics for the inferring of

underlying probability distribution afterwards.

For the convenience, the followed notation is defined to clarify the relations between the spatial templates. Given a template $\mathbf{T} = \{\mathbf{u}_0, \mathbf{u}_0 + \mathbf{h}_1, \ldots, \mathbf{u}_0 + \mathbf{h}_N\}$ as a set of locations with the center node denoted as $center(\mathbf{T}) = \mathbf{u}_0$, the size of the $\mathbf{T}$ is the same as the number of the elements in it and is denoted as $|\mathbf{T}|$, i.e., $|\mathbf{T}| = N + 1$ here. Since the replicates of the data events are matched by their relative positions to the center node regardless of the location of the center node, the relations between the spatial templates are defined in the same manner. Let $\mathbf{T}_a = \{\mathbf{u}_a, \mathbf{u}_a + \mathbf{h}_1, \ldots, \mathbf{u}_a + \mathbf{h}_{N_a}\}$ and $\mathbf{T}_b = \{\mathbf{u}_b, \mathbf{u}_b + \mathbf{h}_1, \ldots, \mathbf{u}_b + \mathbf{h}_{N_b}\}$ be the two spatial templates under consideration, then the relations between $\mathbf{T}_a$ and $\mathbf{T}_b$ are the following:

(1) If $|\mathbf{T}_a| = |\mathbf{T}_b|, \forall t_a \in \mathbf{T}_a, \exists!\ t_b \in \mathbf{T}_b$, such that $t_a - center(\mathbf{T}_a) = t_b - center(\mathbf{T}_b)$, then $\mathbf{T}_a$ and $\mathbf{T}_b$ have the same geometry configuration and the identical relation is expressed as $\mathbf{T}_a = \mathbf{T}_b$.

(2) If $|\mathbf{T}_a| \leq |\mathbf{T}_b|, \forall t_a \in \mathbf{T}_a, \exists!\ t_b \in \mathbf{T}_b$, such that $t_a - center(\mathbf{T}_a) = t_b - center(\mathbf{T}_b)$, then $\mathbf{T}_b$ contains the geometry configuration as a subset and the relation is expressed as $\mathbf{T}_a \subseteq \mathbf{T}_b$ or $\mathbf{T}_b \supseteq \mathbf{T}_a$. If $|\mathbf{T}_a| < |\mathbf{T}_b|$ strictly, the above relation is expressed as $\mathbf{T}_a \subset \mathbf{T}_b$ or $\mathbf{T}_b \supset \mathbf{T}_a$.

Suppose that the spatial template of the conditioning data is $\mathbf{T} = \{\mathbf{u}_0, \mathbf{u}_0 + \mathbf{h}_1, \ldots, \mathbf{u}_0 + \mathbf{h}_N\}$ and that the nodes are ordered increasingly according to their distances from the center. By dropping the furthest node from the template $\mathbf{T}$ each time, a hierarchical set of spatial templates can be defined as

$$v_N = \mathbf{T} \supseteq v_{N-1} = \mathbf{T}\ \{\mathbf{u}_0 + \mathbf{h}_N\} \supseteq, \ldots, \supseteq v_1 = \{\mathbf{u}_0, \mathbf{u}_0 + \mathbf{h}_1\} \supseteq v_0 = \{\mathbf{u}_0\}, \tag{6.2}$$

and the corresponding sets of random variables as

$$V_0 = \{Z_0\} \subseteq V_1 = \{Z_0, Z_1\} \subseteq, \ldots, \subseteq V_N = \{Z_0, Z_1, \ldots, Z_N\}. \tag{6.3}$$

These spatial templates consist of the possible spatial configurations of the partially matched replicates considered in this paper and the entire set is denoted as $G = \cup_{i=1}^{N} v_i$. Let the training data from the replicates associated with the G be denoted as $\mathcal{G}$. For any spatial template $v \in G$, the set of random variables associated with v is denoted as V and the replicates corresponding to the spatial template v is noted as $\mathcal{G}_v$. The size of the set $\mathcal{G}_v$ is noted as $|\mathcal{G}_v|$ representing the number of replicates associated with the spatial template v. And let the total number of replicates associated with G be $|\mathcal{G}|$. An arbitrary element

$\boldsymbol{\zeta}_{t,v} \in \mathcal{G}_v$, represents a sequence of attribute values as

$$\boldsymbol{\zeta}_{t,v} = \{\zeta_{t,i} : i \in v\}, \tag{6.4}$$

where $\zeta_{t,i}$ are the values from the replicate at the location of node i in the spatial template v and $1 \leq t \leq |\mathcal{G}_v|$ corresponds to one of the replicates. The element mapped to the corresponding kernel subspace from $\boldsymbol{\zeta}_{t,v}$ can be represented as

$$\kappa[\boldsymbol{\zeta}_{t,v}] = K_V(\boldsymbol{\zeta}_{t,v}, \cdot), \tag{6.5}$$

which is a function element in the kernel space. With the replicates in $\mathcal{G}_v$ mapping to the kernel space with kernel K_V, the empirical kernel mean $\kappa[\mathcal{G}_v]$ can be defined as

$$\kappa[\mathcal{G}_v] = \frac{1}{|\mathcal{G}_v|}\sum_{t=1}^{|\mathcal{G}_v|}\kappa[\boldsymbol{\zeta}_{t,v}] = \frac{1}{|\mathcal{G}_v|}\sum_{t=1}^{|\mathcal{G}_v|}K_V(\boldsymbol{\zeta}_{t,v}, \cdot). \tag{6.6}$$

For any two nodes $v, v' \in G$ and $v' \supseteq v$, there would be a hereditary subset of replicates which are generated from the projection of v' onto v by restricting the training data $\mathcal{G}_{v'}$ to the spatial template v, and denote this hereditary subset as $\mathcal{G}_{v'|v}$. Obviously, $\mathcal{G}_{v'|v} = \mathcal{G}_v$ if $v' = v$. Given that $v' \supseteq v$, the projected elements in the original data space, their mapped elements in the kernel spaces and the kernel statistics can be defined similarly as

$$\boldsymbol{\zeta}_{t,v'|v} = \{\boldsymbol{\zeta}_{t,i} : i \in v'|v, 1 \leq t \leq |\mathcal{G}_{v'}|\}, \tag{6.7}$$

$$\kappa[\boldsymbol{\zeta}_{t,v'|v}] = K_V(\boldsymbol{\zeta}_{t,v'|v}, \cdot), \tag{6.8}$$

$$\kappa[\mathcal{G}_{v'|v}] = \frac{1}{|\mathcal{G}_{v'}|}\sum_{t=1}^{|\mathcal{G}_{v'}|}\kappa[\boldsymbol{\zeta}_{t,v'|v}] = \frac{1}{|\mathcal{G}_{v'}|}|\sum_{t=1}^{|\mathcal{G}_{v'}|}K_V(\boldsymbol{\zeta}_{t,v'|v}, \cdot). \tag{6.9}$$

Then, the aggregated kernel statistics $\kappa[\mathcal{G}]$ based on the replicates associated to the ensemble of various spatial templates in G can be defined as

$$\kappa[\mathcal{G}] = \sum_{n=1}^{N}\frac{1}{\sum_{i=n}^{N}|\mathcal{G}_{v_i}|}\cdot\Big(\sum_{i=n}^{N}\Big(\kappa[\mathcal{G}_{v_i|v_n}] - \kappa[\mathcal{G}_{v_i|v_{n-1}}]\Big)|\mathcal{G}_{v_i}|\Big). \tag{6.10}$$

Combined with Eq. (6.6), it can be also written as

$$\kappa[\mathcal{G}] = \sum_{n=1}^{N} \frac{1}{\sum_{i=n}^{N} |\mathcal{G}_{v_i}|} \cdot \Big(\sum_{i=n}^{N} \sum_{t=1}^{|\mathcal{G}_{v_i}|} \Big[K_{V_n}(\boldsymbol{\zeta}_{t,v_i|v_n}, \cdot) - K_{V_{n-1}}(\boldsymbol{\zeta}_{t,v_i|v_{n-1}}, \cdot) \Big] \Big). \tag{6.11}$$

6.3.2 Sequential simulation via statistical learning with aggregated kernel statistics

The general concept of statistical learning refers to learning any functional dependency from certain data set without prior knowledge of the data [110, 134]. Herein, the statistical learning framework for the high-order sequential simulation, specifically, means to learn the conditional probability distribution based on the observed replicates from the sample data. The learning procedure can be achieved conveniently through an optimization algorithm in the SLM-kernel space. In fact, the kernel mean defines a feature map to embed probability distribution to the associated kernel space [135, 136, 138]. The empirical mean in the kernel space embeds the empirical probability distribution. Similarly, the expected mean in the kernel space given a certain probability distribution embeds the distribution as an element in the kernel space. Minimizing the distance between the two above-mentioned elements in the kernel space leads to matching of high-order spatial statistics of the target distribution to those of the available data with the kernel space defined by the SLM-kernel.

Equation (6.11) defines a feature map through the aggregated kernel statistics from an ensemble of kernel subspaces. Suppose that the conditioning data is $\Lambda = \{\zeta_1, \ldots, \zeta_N\}$, and define the conditioned kernel statistics $\kappa[\mathcal{G}; \Lambda]$ as

$$\kappa[\mathcal{G}; \Lambda] = \sum_{n=1}^{N} \frac{1}{\sum_{i=n}^{N} |\mathcal{G}_{v_i}|} \cdot \Big(\sum_{i=n}^{N} \sum_{t=1}^{|\mathcal{G}_{v_i}|} \Big[K_{V_n}(\boldsymbol{\zeta}_{t,v_i|v_n}, \Lambda) - K_{V_{n-1}}(\boldsymbol{\zeta}_{t,v_i|v_{n-1}}, \Lambda) \Big] \Big) \tag{6.12}$$

Furthermore, marginalization of $\kappa[\mathcal{G}; \Lambda]$ can be defined as

$$\kappa[\mathcal{G}|\Lambda] = \frac{\kappa[\mathcal{G}; \Lambda]}{\int_{[-1,1]} \kappa[\mathcal{G}; \Lambda] dz_0}. \tag{6.13}$$

The emphasis herein, is to derive a feasible computational model for the marginalized kernel statistics, $\kappa[\mathcal{G}|\Lambda]$, defined in Eq. (6.13). An interesting property of SLM-kernel from its definition is

$$K_{V_i} = K_{V_i \setminus V_{i-1}} K_{V_{i-1}}, i = 1, \ldots, N, \tag{6.14}$$

which means the high-order dimensional kernels could be built incrementally from the lower-dimensional ones as

$$K_{V_N} = K_{V_N \setminus V_{N-1}} K_{V_{N-1} \setminus V_{N-2}} \cdots K_{V_2 \setminus V_1} K_{V_1}. \tag{6.15}$$

Obviously, $V_i \backslash V_{i-1} = \{Z_i\}$ is a single element set and the kernel $K_{V_i \backslash V_{i-1}}$ can be written as

$$K_{V_i \backslash V_{i-1}}(x_i, y_i) = \sum_{w=0}^{W} (w + \frac{1}{2}) P_w(x_i) P_w(y_i). \tag{6.16}$$

Note the orthogonal property of Legendre polynomials, it is easy to derive that

$$\int_{[-1,1]} K_{V_0}(z_0, \cdot) dz_0 = 1, \tag{6.17}$$

and therefore, there is

$$\int_{[-1,1]} \kappa[\mathcal{G}; \Lambda] dz_0 = \sum_{n=1}^{N} \frac{1}{\sum_{i=n}^{N} |\mathcal{G}_{v_i}|} \cdot \Big(\sum_{i=n}^{N} \sum_{t=1}^{|\mathcal{G}_{v_i}|} \Big[K_{V_n \backslash V_0}(\boldsymbol{\zeta}_{t, v_i | v_n}, \Lambda) - K_{V_{n-1} \backslash V_0}(\boldsymbol{\zeta}_{t, v_i | v_{n-1}}, \Lambda) \Big] \Big). \tag{6.18}$$

According to Eq. (6.16), the result of Eq. (6.19) can be obtained from the intermediate result of computing Eq. (6.12). In the end, $\kappa[\mathcal{G}|\Lambda]$ an be expressed in the form as

$$\kappa[\mathcal{G}|\Lambda] = \sum_{t=1}^{|G|} \beta_t K_{V_0}(\boldsymbol{\zeta}_{t,0}, z_0), \tag{6.19}$$

where β_t are constant coefficients that can be computed through Eqs. (6.13) and (6.18). Equation (6.19) is a linear combination of elements in kernel space determined by kernel K_{V_0}, and therefore marginalization of the aggregated kernel statistics, $\kappa[\mathcal{G}|\Lambda]$, embeds the empirical conditional probability distribution to the corresponding kernel space with kernel K_{V_0}. Given a convex space P_0 as the solution space of the target distribution $\hat{p}$ and consider the training data replicates in $\mathcal{G}$, the two elements embedding into the kernel space H associated to kernel K_{V_0} are represented as $\mu_{K_{V_0}}[\mathcal{G}]$ and $\mu_{K_{V_0}}[\hat{p}]$, corresponding to the empirical distribution and the target distribution, respectively. The target conditional probability distribution $\hat{p}$ can be solved by the below minimization problem as

$$\min_{\hat{p}} \parallel \mu_{K_{V_0}}[\mathcal{G}] - \mu_{K_{V_0}}[\hat{p}] \parallel_{\mathcal{H}}^2 . \tag{6.20}$$

The minimization in Eq. (6.20) can be expanded to a quadratic programming problem by noticing that the inner products can be expressed as kernel functions. The details to solve the problem given $\hat{p}$ as a convex combination of certain prototype distributions, is established in Yao et al. [148] and thus will not be repeated here. It should be noted that although Eq. (6.19) appears in a similar form as Eq. (16) in Yao et al. [148], the coefficients β_t in Eq. (6.19) depend on the aggregated kernel statistics with different spatial templates, which is

critical for the utilization of information from partially matched replicates.

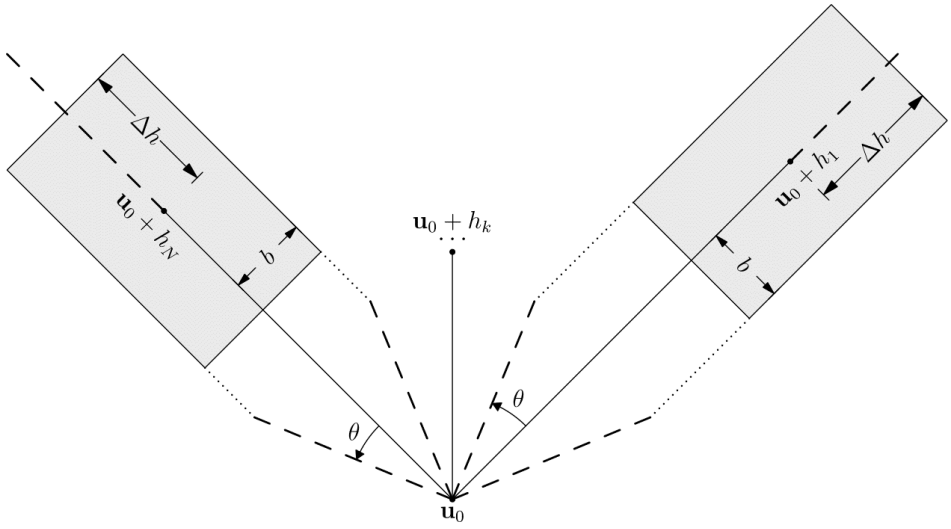


Figure 6.1 Tolerances along each distance vector of the spatial template for retrieving replicates from the samples

With the computation of aggregated kernel statistics of various spatial templates and the auxiliary procedure to estimate the conditional probability distribution, the sequential simulation method via statistical learning with aggregated kernel statistics can be described as the following:

(1) Transform the sample data to the interval [-1, 1] of Legendre polynomials.

(2) Initialize a random path to visit the simulation grid.

(3) For each node to be simulated, find the conditioning data as the data event. The nodes from the spatial template of the data event are ordered increasingly from their distances to the center node.

(4) For each distance vector in the spatial template, allow certain angle tolerance θ and lag tolerance Δh as well as a bandwidth b to find matched node from the samples (Fig. 6.1). Start from the distance vector nearest to the center node and go through all the distance vectors orderly until no matching node is found from the samples. Scan the entire sample data set and store the replicates to separate lists according to the number of nodes matched to the spatial template of the data event.

(5) Compute the aggregated kernel statistics from the partially matched replicates retrieved in Step (4) following Eq. (6.11) and (6.12).

(6) Compute the marginalized kernel statistics defined by Eq. (6.13) and the feature map $\kappa[\mathcal{G}|\Lambda]$ defined by Eq. (6.19), solve the minimization problem in Eq. (6.20) to get an estimated conditional probability distribution. Draw a random sample from the estimated probability distribution and add the value to the simulation grid.

(7) Repeat from Step (3) until all the nodes of the simulation grid are visited.

(8) Back transform the simulate grid from the interval [-1, 1] to generate a realization in the original data space.

6.4 Case Study with a Synthetic Data Set

The synthetic data is a horizontal section extracted from a fully known reservoir data set of porosity [129]. Two different sample data set are drawn from the section representing different sampling density. The data set DS-1 contains samples randomly drawn from 200 locations and the data set DS-2 has 400 samples with regular spacing. Figure 6.2 shows the samples and Fig. 6.3 displays the exhaustive image.

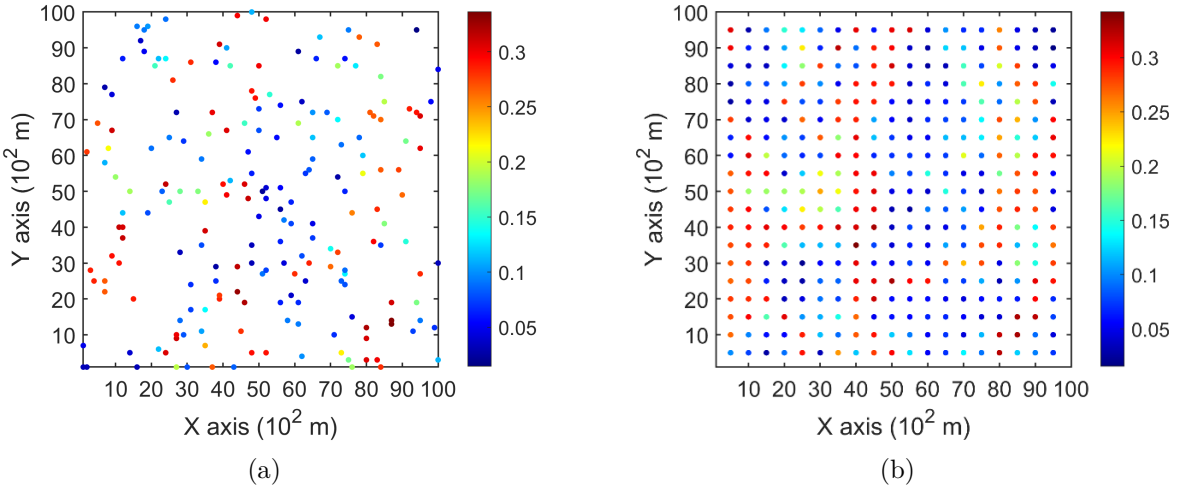


Figure 6.2 Two different sample data set. a DS-1 with 200 randomly drawn samples, b DS-2 with 400 samples

Two realizations of the proposed high-order simulation method using DS-1 and DS-2 is demonstrated in Fig. 6.4a, b and Fig. 6.4c, d respectively. The same random paths are used to for the two realizations for comparison of the impact of sampling density on the simulation method. The visual comparison with the exhaustive image shows that both realizations reproduce the preferential channels along the vertical direction. This shows that the proposed method has the generalization capacity to provide stability of simulation with

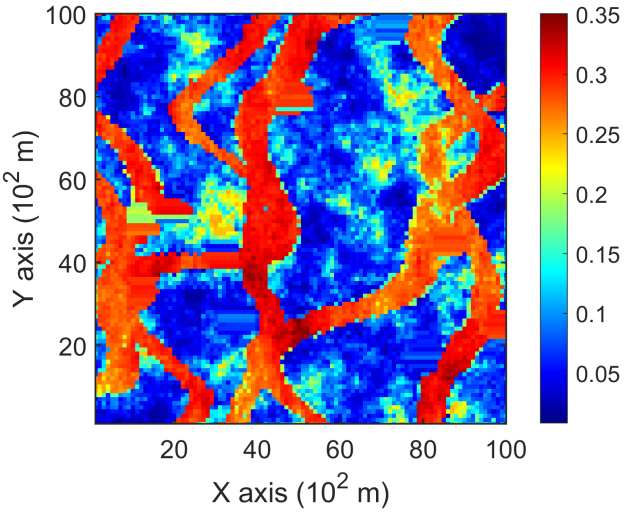


Figure 6.3 A horizontal section of porosity attribute from a reservoir, acting as the exhaustive image

relatively sparse data. On the other hand, the realizations using DS-2 as the sample data retains more fine structures as well as the overall spatial connectivity than the other realization. The reason is that sparser data set in general has less replicates for small structures and, thus, the estimated high-order spatial statistics have to be generalized to stabilize the statistical inference in the situation that the replicates are less. Generally speaking, as the amount of data increases, the models tend to have more variations in finer spatial structures and vice versa.

To further demonstrate the TI-free feature of the proposed simulation method, two realizations of the high-order simulation based on statistical learning using a TI from Yao et al. [148] are displayed in Fig. 6.4e, f for comparison. The results show that the TI adds complementary information to finer structures of the realizations, however, the additional information from the TI seems less compliant to the ground truth when the samples are relatively sparse. As the samples are relatively dense, the contribution of the additional information from the TI also becomes less important since the TI-free simulation method can generate more details from the available sample data. The comparison of histograms of 10 realizations with DS-1 and DS-2 as the sample data with the histograms of the two sample data sets, as well as the exhaustive image, is demonstrated in Fig. 6.5a, b. In both cases, the histograms of the realizations follow the histograms of the sample data sets, whereas the one with dense data resembles more to the exhaustive image, as expected.

The variograms of 10 realizations based on the proposed simulation method using the two different sample data sets are shown in Fig. 6.6, showing that the simulations reproduce

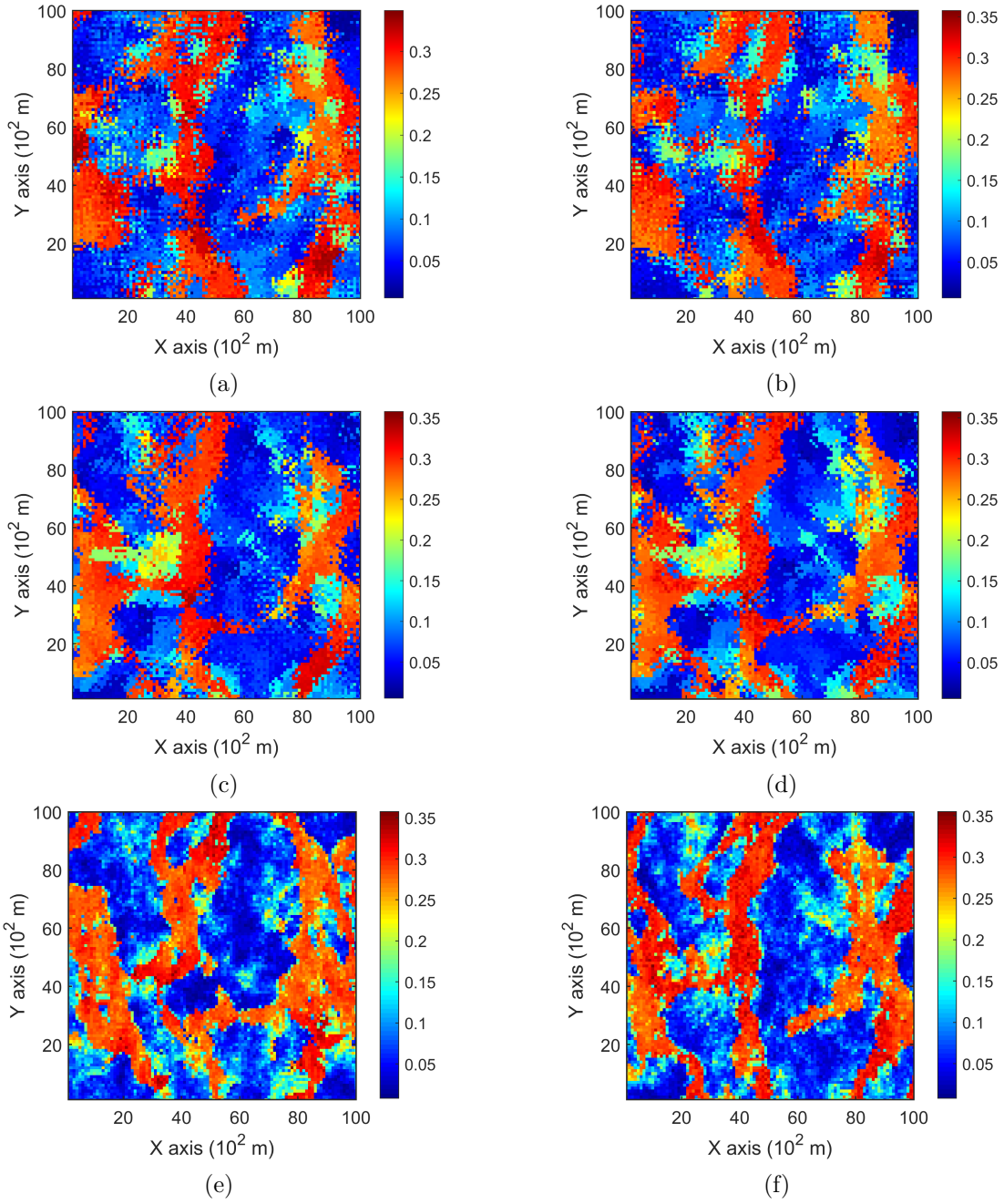
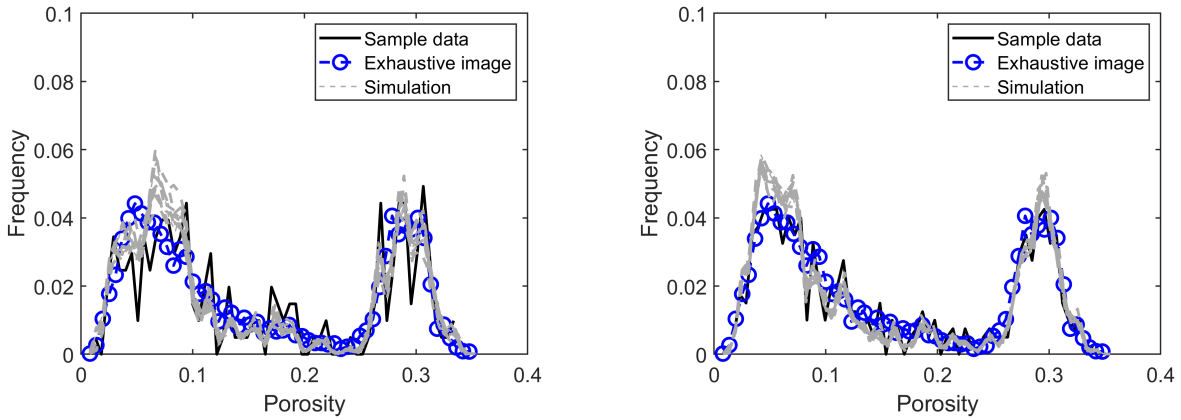


Figure 6.4 Realizations of TI-free high-order simulation with the sample data DS-1 in a, b and with the sample data DS-2 in c, d; for comparison, realizations of high-order simulation using a TI with the sample data DS-1 in e and with the sample data DS-2 in f (from [148])



(a) (b)

Figure 6.5 Histograms of the sample data, the exhaustive image and 10 realizations using a DS-1 and b DS-2 as the sample data, respectively

the variograms of the samples. The third-order cumulant maps of the sample data and the corresponding realizations with the proposed simulation method are shown in Fig. 6.7. Furthermore, the fourth-order cumulate maps of the sample data and the realizations are displayed in Fig. 6.8 for comparison. In this example, the third-order cumulant maps are calculated based on a spatial template along X and Y axes with varied lengths on both directions. The spatial templates of the fourth-order cumulants include extra distance vectors along the diagonal direction in addition to the two axes directions. The fourth-order cumulant maps are also scaled by their deviations for better contrast of the patterns. In general, these high-order cumulant maps represent more complex spatial patterns which characterize interrelations among multiple points. The cumulant maps of two representative realizations from the high-order simulation based on statistical learning using a TI are displayed in the bottom of Fig. 6.7 and Fig. 6.8 for comparison with the results from the proposed method. The comparisons of the cumulant maps suggest that the proposed method is able to reproduce the high-order spatial statistics of the sample data as well as the exhaustive image. The results above show that the proposed approach leads to a reliable inference on the underlying random field model, given a reasonable number of samples available and thus avoids the potential statistical conflicts using a TI to carry out the high-order simulation.

6.5 Conclusions

This paper presents a high-order sequential simulation approach based on statistical learning with aggregated kernel statistics from a set of sample data. Regarding the sparsity of the sample data used to infer the high-order spatial statistics of the underlying random field

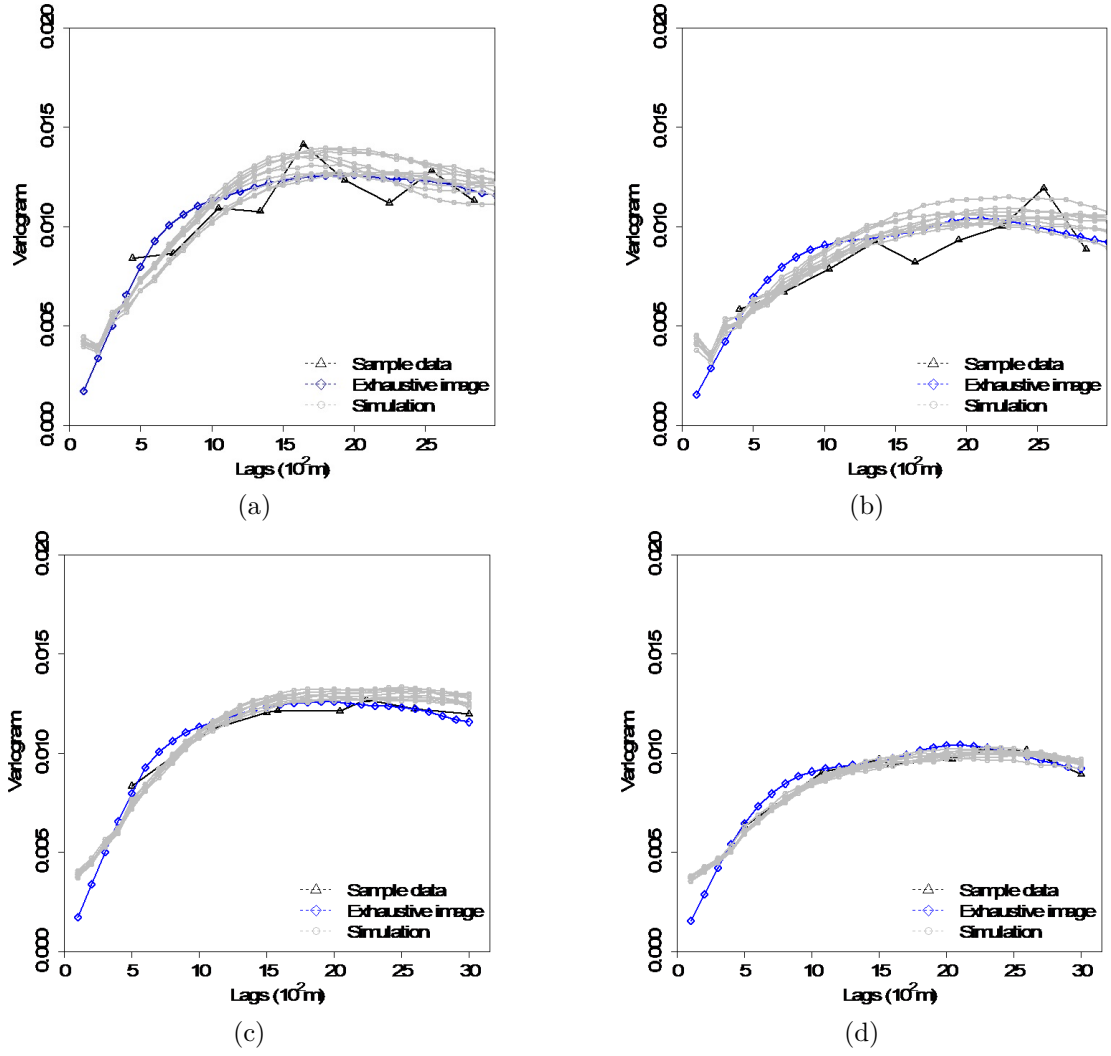


Figure 6.6 Variograms of 10 realizations. a and b, along X and Y axis with DS-1 as the sample data; c and d, along X and Y axis with DS-2 as the sample data

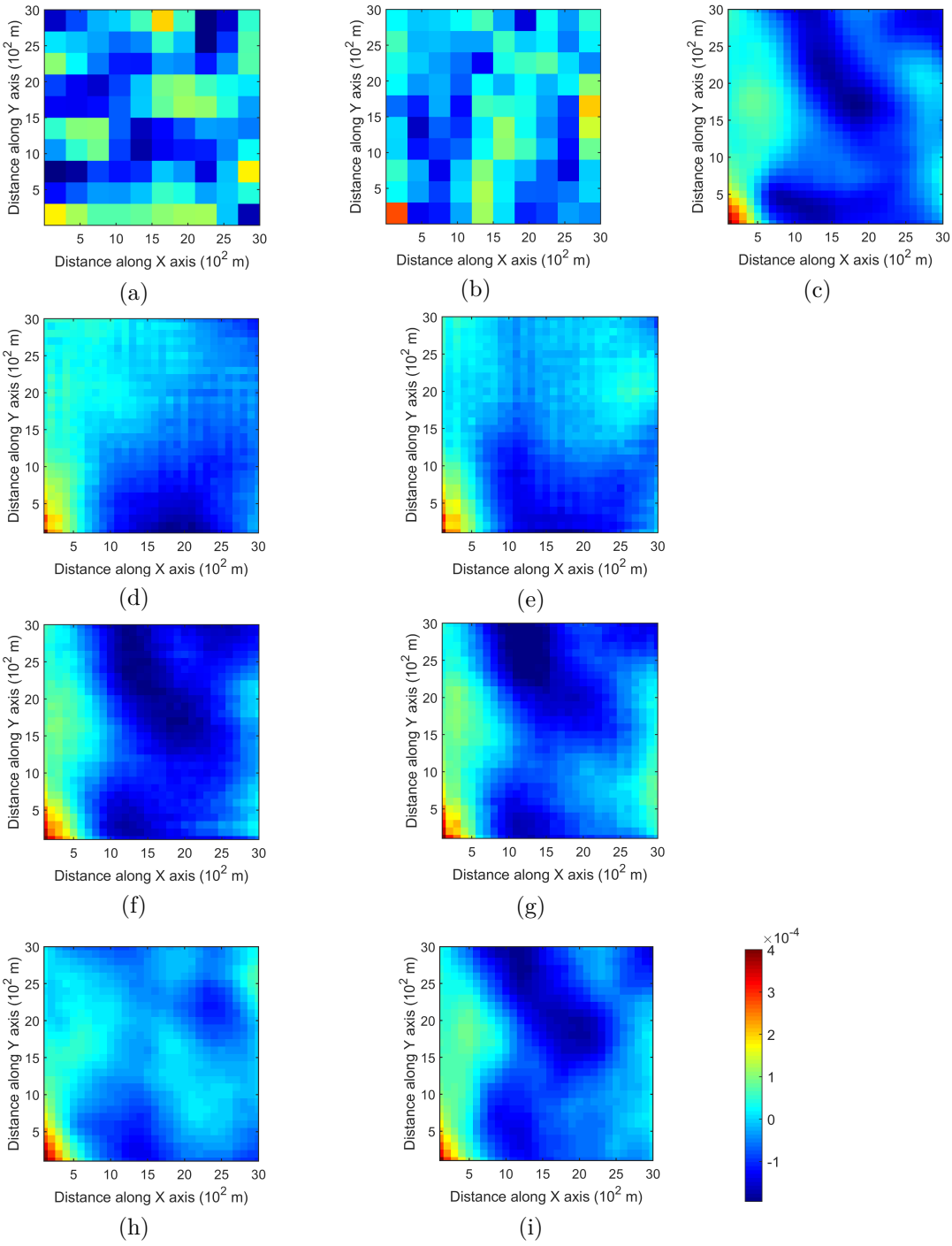


Figure 6.7 Third-order cumulant maps of a DS-1, b DS-2, c exhaustive image, d, e realizations in Fig. 4a, b with DS-1 as the sample data, f, g realizations in Fig. 4c, d with DS-2 as the sample data, h, i realizations of high-order simulation using a TI with DS-1 and DS-2 as the sample data, respectively (from [148])

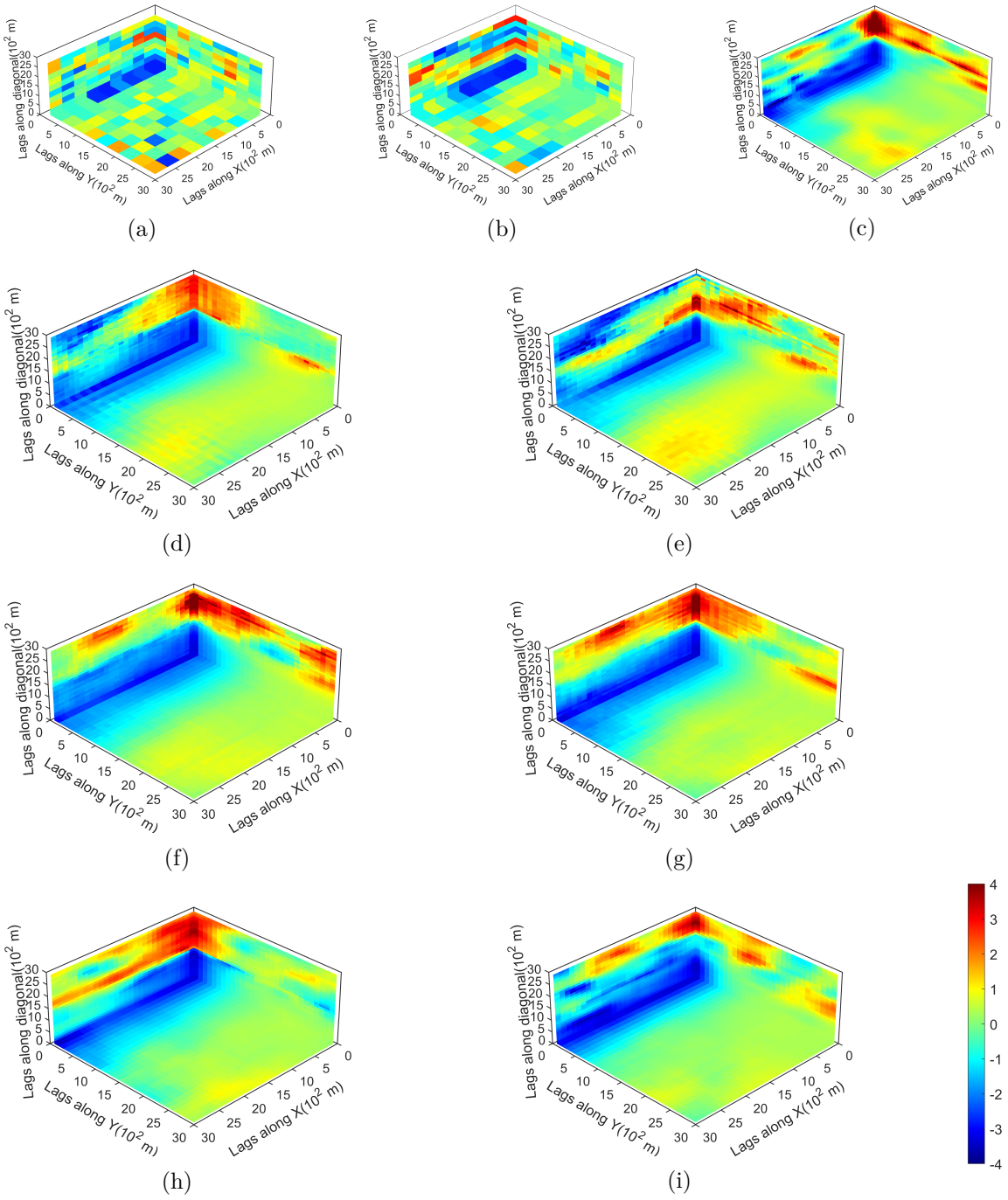


Figure 6.8 Fourth-order cumulant maps of a DS-1, b DS-2, c exhaustive image, d, e realizations in Fig. 4a, b with DS-1 as the sample data, f, g realizations in Fig. 4c, d with DS-2 as the sample data, h, i realizations of high-order simulation using a TI with DS-1 and DS-2 as the sample data, respectively (from [148])

model, the partially matched replicates of the data events encountered in the simulation are mapped into kernel subspaces. The latter kernel subspaces are defined by different kernel functions corresponding to different configurations of spatial templates to create an ensemble set of elements in kernel subspaces. The ensemble of elements in the kernel subspaces are aggregated to construct the new concept of aggregated kernel statistics. The aggregated kernel statistics are crucial in building a new feature map to consider partially matched replicates together to the same kernel space of the conditional probability distribution. In addition, the statistical learning framework for high-order simulation offers the generalization capacity for sparse data learning. The combination of the aggregated kernel statistics with the statistical learning thus provides a new way to derive the proposed TI free high-order simulation method. The proposed method tackles the issue of statistical conflicts between the sample data and the TI. The case study from the fully-known data set shows that the proposed method reproduces both lower-order and higher-order spatial statistics in generated realizations. Even with relatively sparse samples, the proposed method retains the main spatial patterns of the available data, which is characterized by high-order spatial statistics. It should be noted that the concept of aggregated kernel statistics is quite flexible and can accommodate information from different data sources with various spatial configurations. This represents a potential direction for future research.

CHAPTER 7 ARTICLE 4: LEARNING HIGH-ORDER SPATIAL STATISTICS AT MULTIPLE SCALES: A KERNEL-BASED STOCHASTIC SIMULATION ALGORITHM AND ITS IMPLEMENTATION

Abstract: This paper presents a learning-based stochastic simulation method that incorporates high-order spatial statistics at multiple scales from sources with different resolutions. Regarding the simulation of a certain spatial attribute, the high-order spatial information from different sources is encapsulated as aggregated kernel statistics in a spatial Legendre moment kernel space, and the probability distribution of the underlying random field model is derived by a statistical learning algorithm, which matches the high-order spatial statistics of the target model to the observed ones. In addition, a related software is developed as the SGeMS plugin. Case studies are conducted with a known data set and a gold deposit, demonstrating reproduction of high-order spatial statistics from the available data, as well as practical aspects in mining applications.

Keywords: Stochastic simulation; High-order spatial statistics; High-order simulation software; Kernel; Statistical learning

7.1 Introduction

High-order stochastic simulation methods are amongst the latest developments in geostatistical simulation, aiming to reproduce complex spatial patterns from the available data. The spatial patterns represent the interaction of spatial attributes of certain natural phenomena among multiple locations and they can be characterized by the high-order spatial statistics defined in different ways such as high-order spatial cumulants or high-order spatial moments [11, 17, 91]. High-order simulation methods contrast with the multiple point simulation approaches, where the multi-point interrelations are indirectly captured as either the frequency of data events occurring at multiple locations [12, 57, 84, 122] or as similarity measures amongst patterns [56, 60, 65–67]. Instead, the high-order simulation methods explicitly build probabilistic models based on high-order spatial statistics. For instance, Legendre polynomial expansion series are used to approximate the probability distributions of spatial attributes where the expansion coefficients are determined by computing the spa-

Submitted: Yao, L., Dimitrakopoulos, R., and Gamache, M. (2020). Learning high-order spatial statistics at multiple scales: a kernel-based stochastic simulation algorithm and its implementation. Computers & Geosciences. (Submitted).

tial cumulants, leading to an early development of a high-order simulation algorithm known as HOSIM [18, 19]. The concept of high-order spatial statistics has also been extended to multiple variables to develop joint simulation of spatially correlated attributes [99]. The probabilistic model of high-order simulation makes no parametric assumptions of the probability distribution and thus characterizes the non-gaussian and non-linear features of the spatial attributes. The estimation of the probability distribution yields a numerical model based on components linked to the empirical high-order spatial statistics calculated from the available data. In practice, the input data for estimating the probability distribution may impact the numerical stability of the related estimation. The available sample data alone may not be sufficient to infer the high-order spatial statistics required and, thus, may influence the numerical model. This limitation is alleviated with the use of a training image (TI) as the complementary statistical analog [18]. Another approximation model of a high-order simulation that shows substantial improvement with regards to numerical stability is found in Minniakhmetov et al. [128]. The latter authors use the Legendre-like splines as the basis functions for the approximation series, which leads to a better reproduction of spatial data patterns, as compared to the previous HOSIM method.

A concern when using a TI as a statistical analog of the underlying random field model is the possible statistical conflicts between the sample data and the TI. Yao et al. [148] propose a statistical learning framework for high-order sequential simulation in a newly defined kernel space; the related learning algorithm shows generalization capacity to comply with the inferred model from the TI with the spatial statistics of the sample data, and thereby mitigates the possible statistical conflicts. A spatial Legendre moment kernel is proposed in Yao et al. [148] to define the associated kernel space. The replicates of the data events (conditioning data) retrieved from the available data are mapped into the spatial Legendre moment kernel space by a feature mapping function. Thereafter, the so-called empirical kernel statistics are defined by taking a sample average of the mapped elements in the kernel space corresponding to the replicates. As a result, the empirical kernel statistics carried high-order spatial statistics of the replicates. On the other hand, the target probability distribution from the related random field model can be embedded into the same kernel space through the termed expected kernel statistics. A kernelized learning algorithm is designed specifically to match the expected kernel statistics to the empirical kernel statistics, which results in a simulation model with a reproduction of high-order spatial statistics from the available data. Although the proposed statistical learning framework is general, one limitation of the application in Yao et al. [148] is that the replicates retrieved from the TI act as the only training data in the related learning algorithm; this may influence the spatial continuity of the realizations given that the statistical conflicts between the sample data and the TI are severe. Yao et al. [149]

propose a TI-free high-order simulation method based on the statistical learning framework. The concept of aggregated kernel statistics is defined such that the samples with different spatial configurations can be effectively utilized for statistical inference of the random field model. A limitation of the above-mentioned TI-free simulation method is that the quality of the realizations depends on the sampling density. While the sample data are relatively sparse, the fine-scale spatial structures of the spatial attributes of interest are not well represented. The limitations found in previous simulation methods motivate the present research to propose a new type of aggregated kernel statistics, which aims to incorporate the high-order spatial information at multiple scales. Specifically, the sample data are relatively sparse and thus carry high-order spatial information at coarse scales. On the other hand, the TIs are exhaustive and can provide high-order spatial information at finer scales. The general idea of the proposed aggregated kernel statistics in this paper is to exclude the influence of the TI from deriving the high-order spatial statistics at coarse scales by only utilizing the sample data, while complementing the high-order spatial information with TI. Thereafter, the aggregated kernel statistics are utilized in the statistical learning framework for further inference of the random field model. Although the present study considers only two different scales of data as the samples and the TI, the concept of the aggregated kernel statistics proposed herein can be generalized to multiple scales. In addition, a high-order simulation program is developed accordingly and described in this paper. The implementation is written in C++ language and is compatible to the SGeMS software.

In the following sections, Section 7.2 presents the high-order simulation method based on statistical learning and the concept of the aggregated kernel statistics. Section 7.3 describes a kernelized high-order simulation program and its implementation in C++ language. Section 7.4 contains two different case studies with a synthetic data set and at a gold deposit. Conclusions are presented in Section 7.5.

7.2 Method

In this section, concepts of high-order sequential simulation are first outlined, followed by a brief overview of the spatial Legendre moment kernel space. The concept of aggregated kernel statistics at different scales is then presented and utilized to develop a kernelized learning algorithm.

7.2.1 High-order sequential simulation

Suppose that the attributes of interest are modeled as a random field $\mathbf{Z}(\mathbf{u})$ where $\mathbf{u}$ represents locations at a certain spatial domain. The attributes at multiple locations within the spatial domain comprise a multivariate probability distribution. The multivariate probability distribution can be decomposed into a sequence of conditional probability distributions so that the random values can be sequentially drawn from the multivariate probability distribution to generate the simulated realizations. Without loss of generality, the conditional probability density functions (CPDF) can be approximated as $f(z_0|\zeta_1, \ldots, \zeta_N)$ given that the node Z_0 to be simulated center at $\mathbf{u}_0$ and the conditioning data within its neighborhood located at $\mathbf{u}_1, \ldots, \mathbf{u}_N$ with the value of attributes corresponding to $\zeta_1, \ldots, \zeta_N$. In terms of high-order sequential simulation, the high-order spatial statistics are taken into account for approximating the CPDF $f(z_0|\zeta_1, \ldots, \zeta_N)$, and the conditioning data $\zeta_1, \ldots, \zeta_N$ are called as a data event associated with a spatial template defined by distance vectors of location $\mathbf{u}_1, \ldots, \mathbf{u}_N$ to the location $\mathbf{u}_0$ of the center node. The high-order spatial statistics are contained in the replicates of a data event for inference. Note that the replicates of a data event in high-order simulation methods are not necessary to have identical or similar attribute values to the data event, but rather to have the same spatial template, i.e., the same data geometry. In general, the replicates from the sample data correspond to spatial template at coarse scales and the replicates from the TI provides spatial information at finer scales because of the sparsity of the sample data in contrast to the exhaustive TI.

7.2.2 Kernel space and spatial Legendre moment kernel

Suppose the original data space of the considered spatial attributes is represented by a nonempty set $\mathbb{E}$, then an element $x \in \mathbb{E}$ can be taken to a kernel space $\mathcal{H}$ by a so-called feature mapping function $\phi(x) : \mathbb{E} \to \mathcal{H}$. The kernel space $\mathcal{H}$ is a Hilbert space with the inner product defined by a positive definite kernel function $K : \mathbb{E} \times \mathbb{E} \to \mathbb{R}$ where $\mathbb{R}$ is the set of the real numbers. Given a Hilbert space $\mathcal{H}$ with the kernel K, then for $x, y \in \mathbb{E}$ and the corresponding features $\phi(x), \phi(y) \in \mathcal{H}$, the inner product on $\mathcal{H}$ can be defined as

$$\langle \phi(x), \phi(y) \rangle_{\mathcal{H}} = K(x, y). \tag{7.1}$$

An interesting property with the kernel K is that the function $\phi(x) : \mathbb{E} \to \mathcal{H}, x \mapsto K(\cdot, x)$ also defines a feature map namely as reproduce kernel map or canonical feature map [111, 140].

The kernel function K has the reproducing property as

$$\langle f(x), K(\cdot, x)\rangle_{\mathcal{H}} = f(x), \tag{7.2}$$

$\forall x \in \mathbb{E}$ and $\forall f \in \mathcal{H}$, therefore there is

$$\langle K(\cdot, x), K(\cdot, y)\rangle_{\mathcal{H}} = K(x, y). \tag{7.3}$$

This kind of reproduce kernel map is adopted throughout this paper as the feature mapping function from the original data space to the kernel space. It is obvious from the Eq. (7.3) that the elements in the kernel space after the feature mapping from the original data space have the similarity measure defined as the distances between each other through the kernel function K.

The spatial Legendre moment kernel [148] allows to carry over the high-order spatial statistics information from the original data space to the newly defined kernel space with the definition as

$$K_V(\mathbf{X}, \mathbf{Y}) = \prod_{i=0}^{N} \Big[\sum_{w=0}^{W} (w + \frac{1}{2}) P_w(x_i) P_w(y_i) \Big], \tag{7.4}$$

where K_V is the kernel corresponding to the set of random variables associated with a spatial template of N distance vectors, and P_w is the Legendre polynomial of order w.

7.2.3 Aggregating kernel statistics at different scales

With the definition of spatial Legendre moment kernel in Eq. (7.4),the empirical kernel statistics can be defined accordingly based on the sample average of the elements in the kernel space mapped from samples in the original data space. The kernel function K_V depends on the spatial template involved, and so as the kernel statistics from the available data are related to the spatial templates of the data events. When both the sample data and the TI are available for retrieving the replicates and inferring the kernel statistics, the replicates from the two different sources generally carry high-order spatial statistics information at different scales. Specifically, the sample data are relatively sparse that frequently the spatial configuration of the replicates from them could only partially match to the spatial template of the data event, and theses replicates carry the spatial statistics at coarser scale with relatively higher compliance to the underlying random field. On the contrary, the TI are exhaustive data and the replicates from it can fully match the spatial template of the data event, thus the replicates provide spatial statistics at finer scale but possibly with less compliance to the underlying random field model.

Suppose that the spatial template of the data event be noted as v and the corresponding set of random variables be noted as V. Let v_s be the spatial template of the replicates of the data event retrieved from the sample data and the associated set of random variables as V_s, v_s and V_s are the subsets of v and V, respectively. Let $\mathcal{G}_{v_s}$ be the set of replicates from the sample data and the number of these replicates be n_s, the kernel statistics based on $\mathcal{G}_{v_s}$ can be defined as

$$\kappa[\mathcal{G}_{v_s}] = \frac{1}{n_s}\sum_{i=1}^{n_s} K_{V_s}(\boldsymbol{\zeta}^s_{i,v_s}, \cdot), \tag{7.5}$$

where $\boldsymbol{\zeta}^s_{i,v_s}$ is the vector of the attribute values corresponding to the replicates in set $\mathcal{G}_{v_s}$. The kernel statistics of the replicates from the TI can be defined separately in a similar way. The motivation of aggregating kernel statistics at different scales is to utilize the part of high-order spatial information of the replicates from the sample data and in the meanwhile complement the rest part of high-order spatial information using the replicates from the TI. In other words, the spatial template v is divided into two sub-templates v_s and v_t respectively corresponding to the sample data and the TI, and so are the set of random variables are divided into V_s and V_t, respectively. Therefore, there are

$$v = v_s \cup v_t, \tag{7.6}$$

and

$$V = V_s \cup V_t. \tag{7.7}$$

The above subdivision regarding the spatial template also leads to kernel subspaces with kernels K_{V_s} and K_{V_t}. Suppose the ensemble of replicates from both the sample data and the TI denote as a set $\mathcal{G}_v$ and let n_t denote the number of replicates from the TI. The aggregated kernel statistics combining the replicates both from the TI and the sample data at different scales are defined as

$$\kappa[\mathcal{G}_v] = \frac{1}{n_s}\sum_{i=1}^{n_s} K_{V_s}(\boldsymbol{\zeta}^s_{i,v_s}, \cdot) + \frac{1}{n_t}\sum_{j=1}^{n_t}[K_V(\boldsymbol{\zeta}^t_{j,v}, \cdot) - K_{V_s}(\boldsymbol{\zeta}^t_{j,v_t}, \cdot)], \tag{7.8}$$

where $\boldsymbol{\zeta}^s_{i,v_s}$, $\boldsymbol{\zeta}^t_{j,v}$ and $\boldsymbol{\zeta}^t_{j,v_t}$ represent the replicates from the sample data and the replicates from the TI with spatial template v and v_t, respectively.

7.2.4 Kernelized high-order sequential simulation algorithm

The high-order spatial information from both the sample and the TI can be represented by the aggregated kernel statistics at two scales. In terms of high-order sequential simulation, the target is to obtain conditional probability distributions which match the high-order spatial statistics of the available data. This matching of high-order spatial statistics can be conveniently achieved by a statistical learning algorithm in kernel space. Suppose that the target probability density function $\hat{p}$ lies in the convex space of certain prototype probability density functions p_i as

$$\hat{p} = \sum_{i=1}^{n} \alpha_i p_i, \tag{7.9}$$

where $\sum_{i=1}^{n} \alpha_i = 1$ and $\alpha_i \geq 0, \forall 1 \leq i \leq n$. It is straightforward that the expected kernel statistics with regard to the probability distribution can be defined as

$$\kappa_0[\hat{p}] = E_{z_0 \sim \hat{p}}[K_0(z_0, \cdot)], \tag{7.10}$$

where Z_0 is the center node to be simulated and K_0 is the corresponding kernel function. The aggregated kernel statistics defined in Eq. (7.8) can be projected to the same kernel space through marginalization, and therefore the expected kernel statistics can be matched to the observed kernel statistics from the available data simply by minimizing the distance of two elements in the kernel space. Given that the conditioning data as $\Lambda = \{\zeta_1, \ldots, \zeta_N\}$ and the evaluation of $\kappa[\mathcal{G}_v]$ on Λ as $\kappa[\mathcal{G}_v|\Lambda]$, the projection of the aggregated kernel statistics can be defined as

$$\kappa_0[\mathcal{G}_v|\Lambda] = \frac{\kappa[\mathcal{G}_v; \Lambda]}{\int_{[-1,1]} \kappa[\mathcal{G}_v; \Lambda] dz_0}. \tag{7.11}$$

Specifically, the statistical learning of high-order spatial statistics leads to a minimization problem

$$\min_{\hat{p}} \parallel \kappa_0[\mathcal{G}_v|\Lambda] - \kappa_0[\hat{p}] \parallel_{\mathcal{H}}^2 . \tag{7.12}$$

The minimization in Eq. (7.12) amounts to solve a quadratic problem in a general form [136] as

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \frac{1}{2} \boldsymbol{\alpha}^T \left(\mathbf{Q} + \lambda \mathbf{I} \right) \boldsymbol{\alpha} - \mathbf{q}^T \boldsymbol{\alpha} \\ &\text{s.t.} \sum_{i=1}^{n} \alpha_i = 1 \\ &\alpha_i \geq 0, \quad \forall 1 \leq i \leq n, \end{aligned} \tag{7.13}$$

where λ is a regularization constant and $\mathbf{I}$ is the identity matrix. Matrix $\mathbf{Q}$ and vector $\mathbf{q}$ differ as the kernel function varies. For deriving the entries of the matrix $\mathbf{Q}$ and vector $\mathbf{q}$, as well as solving the quadratic programming problem in the SLM-kernel space, the readers are referred to Yao et al. [148].

As long as the target conditional probability density functions are determined through the above learning process, the rest of simulation follows the general procedure of sequential simulation. Hence, the kernelized high-order sequential simulation algorithm can be described as follows

(1) Transform sample data and TI to the domain of Legendre polynomials, the interval $[-1, 1]$.

(2) Generate a random path to visit the simulation grid.

(3) Find the conditioning data inside the neighborhood of the current node to simulate as the data event, the spatial template of the data event is used to retrieve replicates from the sample data and the TI.

(4) Compute the aggregated kernel statistics defined in Eq. (7.8) from the replicates retrieved from the sample data and the TI.

(5) Match the kernel statistics of the target CPDF to the aggregated kernel statistics and build the quadratic programming problem through Eq. (7.12) and (7.13). Solve the quadratic programming problem to derive the target CPDF.

(6) Generate a random value from the target CPDF and add it to the simulation grid.

(7) Repeat from steps (3) to (6) until all the nodes on the simulation grid are simulated.

(8) Back transform the node attributes of the simulation from the interval $[-1, 1]$ to the original data space.

7.3 A Kernelized High-Order Simulation Program

The kernelized high-order simulation program is developed as a software plugin compatible with the SGeMS platform [122]. The program is written in C++ language and follows the generic programming paradigm adopted in the design of GsTL, a geostatistical template library [150]. The main workflow contains three major C++ classes which are described as the following.

7.3.1 Class kernelsim

This class is the application class communicating with the SGeMS platform through the user parameters, as well as running the simulation algorithm from the GUI. The class is derived from a predefined interface from the SGeMS platform so that it is compatible to the function calling convention of SGeMS. The object from the class kernelsim calls the sequential simulation function to start the high-order simulation procedure. The parameters of the proposed simulation algorithm can either be input from the GUI by the user or can be loaded from an XML file. The parameters are described in Table 7.1.

Table 7.1 Parameters description

Parameter	Range
Maximum order of Legendre polynomials	between $10 \sim 20$
Maximum number of conditioning data	$10 \sim 30$
Number of replicates from the TI	-1: take all the replicates n>0: n replicates from the TI
Hard data usage	0: only use hard data 1: incorporate both the hard data and the TI -1: not using the hard data (only use the TI)
Angle tolerance	$15° \sim 45°$
Lag tolerance	Application dependent
Bandwidth	Application dependent
Dimensions of searching window	Application dependent
Number of prototype distributions	$10 \sim 20$
Number of divisions on the interval	$100 \sim 200$
Scale parameter of the prototype distribution	$0.01 \sim 0.05$

7.3.2 Class SLM_kde_estimator

This class serves as the role to estimate the conditional probability density function through the learning algorithm. The class SLM_kde_estimator first calls the other function class to process the replicates which returns the aggregated kernel statistics. The main functions inside the class include the selection of the prototype distributions, construction of quadratic programming problem in Eq. (7.13), solving the quadratic programming problem to obtain the target conditional probability density function.

7.3.3 Class replicate_processor

This class is designed for processing the replicates. The conditioning data, the sample data, and the TI are used as input to this class. The spatial templates of the data events are constructed from the spatial configuration of the conditioning data. There are two major member functions defined in this class. The first function retrieves the replicates from both the sample data and the TI, respectively. The other function computes the aggregated kernel statistics from the retrieved replicates according to Eq. (7.8). The aggregated kernel statistics are passed to the object of the class SLM_kde_estimator to estimate the target probability density function.

7.4 Numerical Results

Two separate case studies are carried out to test the developed simulation program. The first case study is conducted with a synthetic data set to verify the performance of the proposed simulation method. The other case study carries out the stochastic orebody modeling at a gold deposit, aiming to test the proposed method in a three-dimensional space, as well as its practical aspects in real-life mines.

7.4.1 Case study with a synthetic data set

The porosity attributes from the Stanford V Reservoir data set [129] are used to conduct the simulation in this case study. Specifically, two sections are extracted from the data set and the sections consist 100×100 cells. One section is regarded as the exhaustive image where 200 points are randomly drawn from this image. The other section is rotated 45° clockwise so that the channels have distinct preferential directions from the exhaustive image after the rotation. The rotated section acts as the TI in this case study to represent the situation of the statistical conflicts existing between the sample data and the TI. The exhaustive image, the TI and the sample data are shown in Fig. 7.1.

Two realizations using the above sample and the TI are displayed in Fig. 7.2. The visualization of the simulated results demonstrates good reproduction of the channels in the preferential orientation along the vertical direction from the exhaustive image. In addition, 10 realizations are generated to evaluate the overall performance of the simulation method in reproducing the low-order statistics. The latter includes the proportions and the second-order spatial statistics from the sample data, where the histograms and variograms of the 10 realizations are compared with those of the sample data, the TI and the exhaustive image, as shown in Fig. 7.3 and Fig. 7.4, respectively. The comparison of the histograms shows

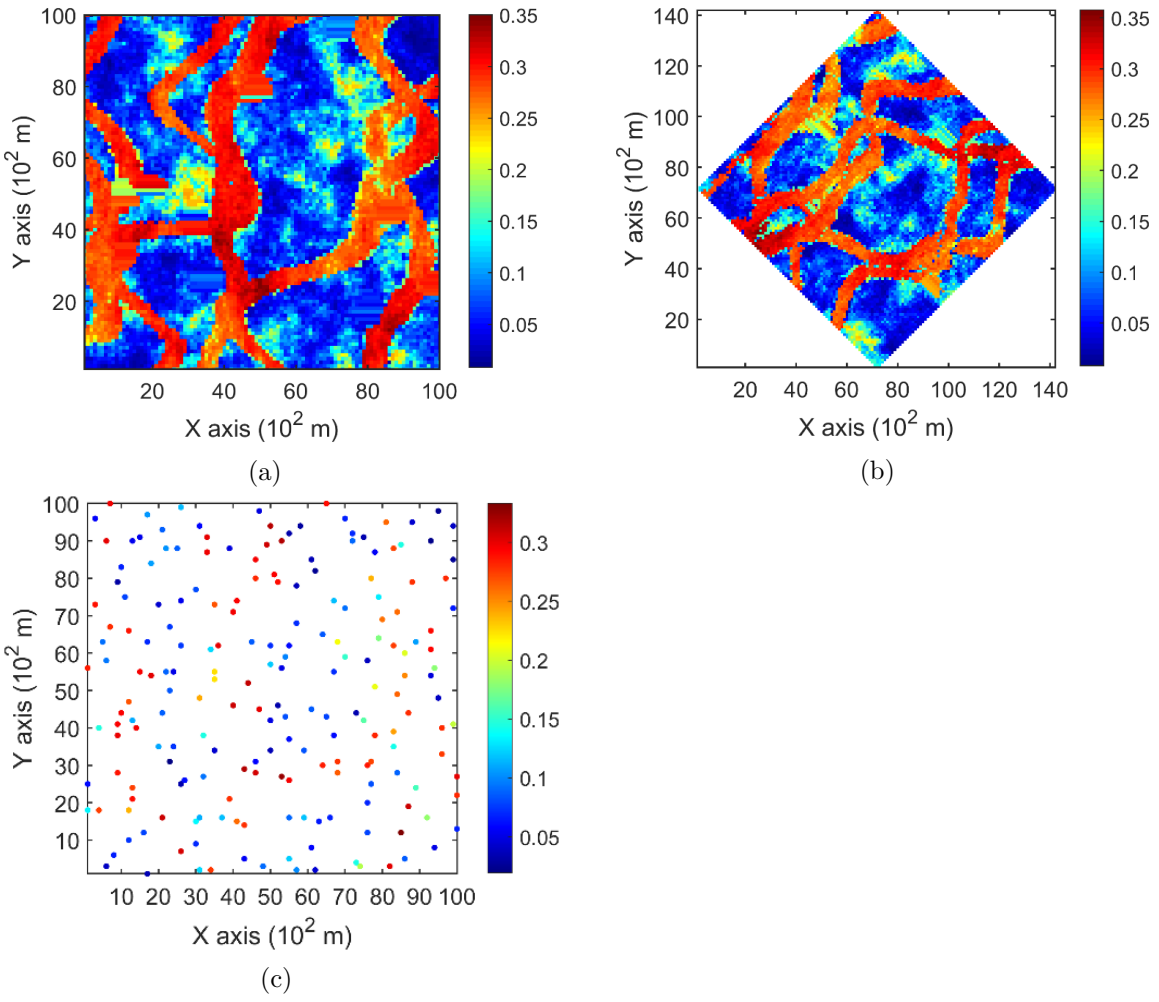


Figure 7.1 a Exhaustive image; b training image; c sample data drawn from the exhaustive image

that the proposed simulation method has a reasonable reproduction of proportions from the sample data as well as the exhaustive image. The comparison of variograms clearly shows that the simulated realizations tend to have the similar second-order spatial statistics to the sample data instead of the TI. A further comparison of the third- and fourth- order cumulant maps of two separate realizations with those of the sample data, the TI and the exhaustive image are illustrated in Fig. 7.5 and Fig. 7.6, respectively. The spatial template used in the third-order cumulant maps includes directions along the X-axis and Y-axis and the spatial template of the fourth-cumulant maps includes an additional direction along the diagonal. The fourth-order cumulant maps are normalized to visually highlight the spatial patterns. Significant difference can be seen between the cumulant maps of the TI and those of the sample data and e exhaustive image. The similarity between the cumulant maps of the realizations and the exhaustive image implies that the simulation method is able to mitigate the statistical conflicts between the samples and the TI, maintaining reasonable reproduction of both low-order and high-order spatial statistics from the sample data.

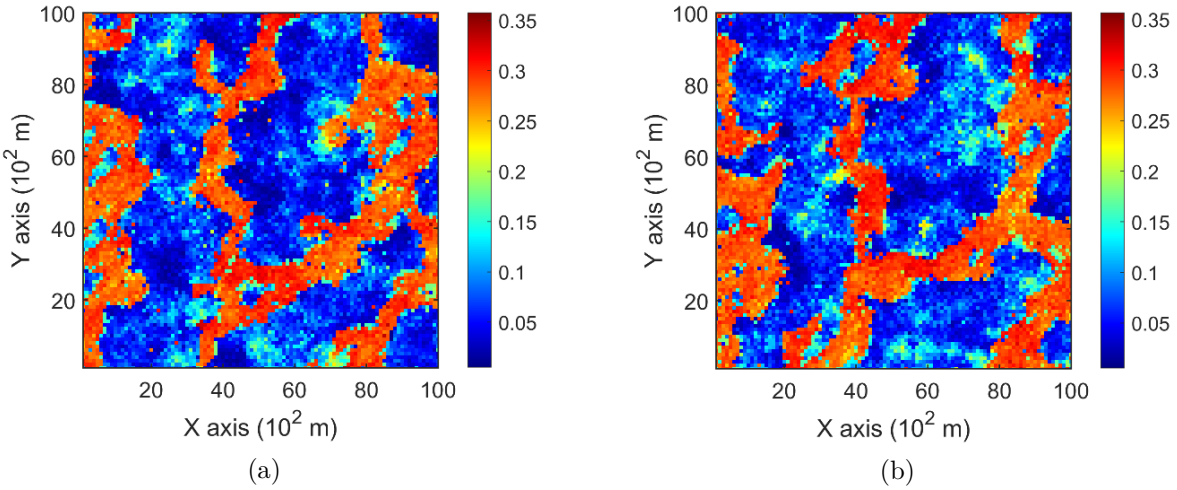


Figure 7.2 Two simulated realizations using the samples and the TI shown in Fig. 7.1

7.4.2 Case study at a gold deposit

The case study at a gold deposit is presented here to demonstrate the practical aspects of the developed simulation program in stochastic orebody modeling. The gold deposit contains samples spatially distributed in 407 exploration drill holes as shown in Fig. 7.7a. The samples are composited to 10 m in length. The simulation grid is defined as blocks of size $5m \times 5m \times 10m$. The TI is generated from blast-hole data in a mined-out area of the deposit, and a cross-section is shown in Fig. 7.7b.

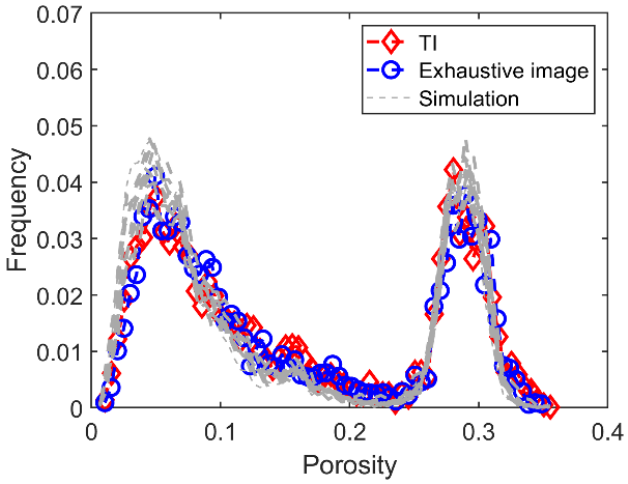
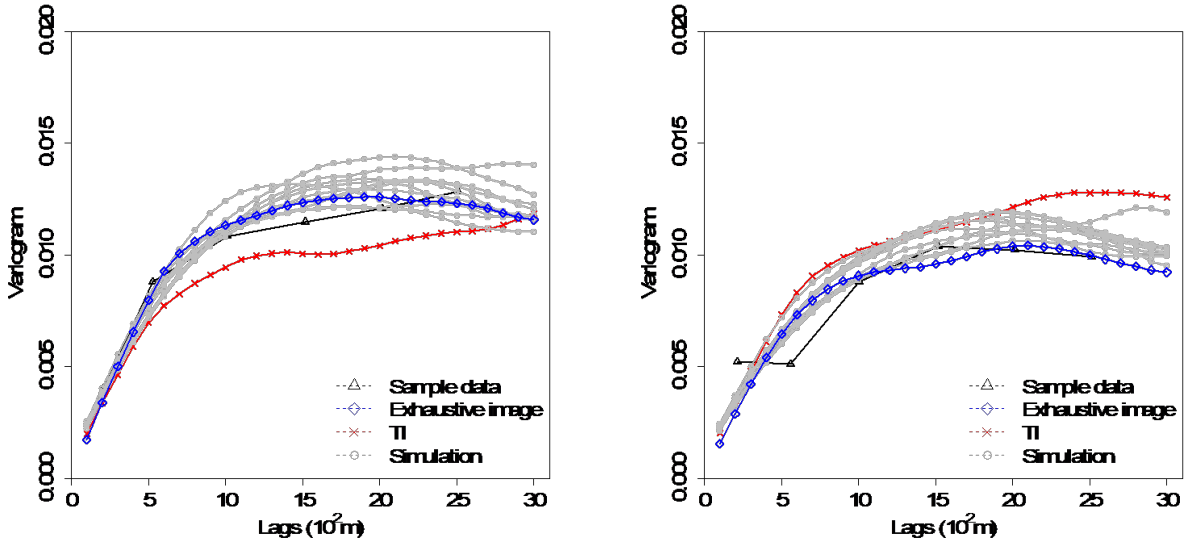


Figure 7.3 Histograms of 10 simulated realizations using the samples and the TI shown in Fig. 7.1



(a) (b)

Figure 7.4 Variograms of 10 simulated realizations along a X-axis and b Y-axis, using the samples and the TI shown in Fig. 7.1

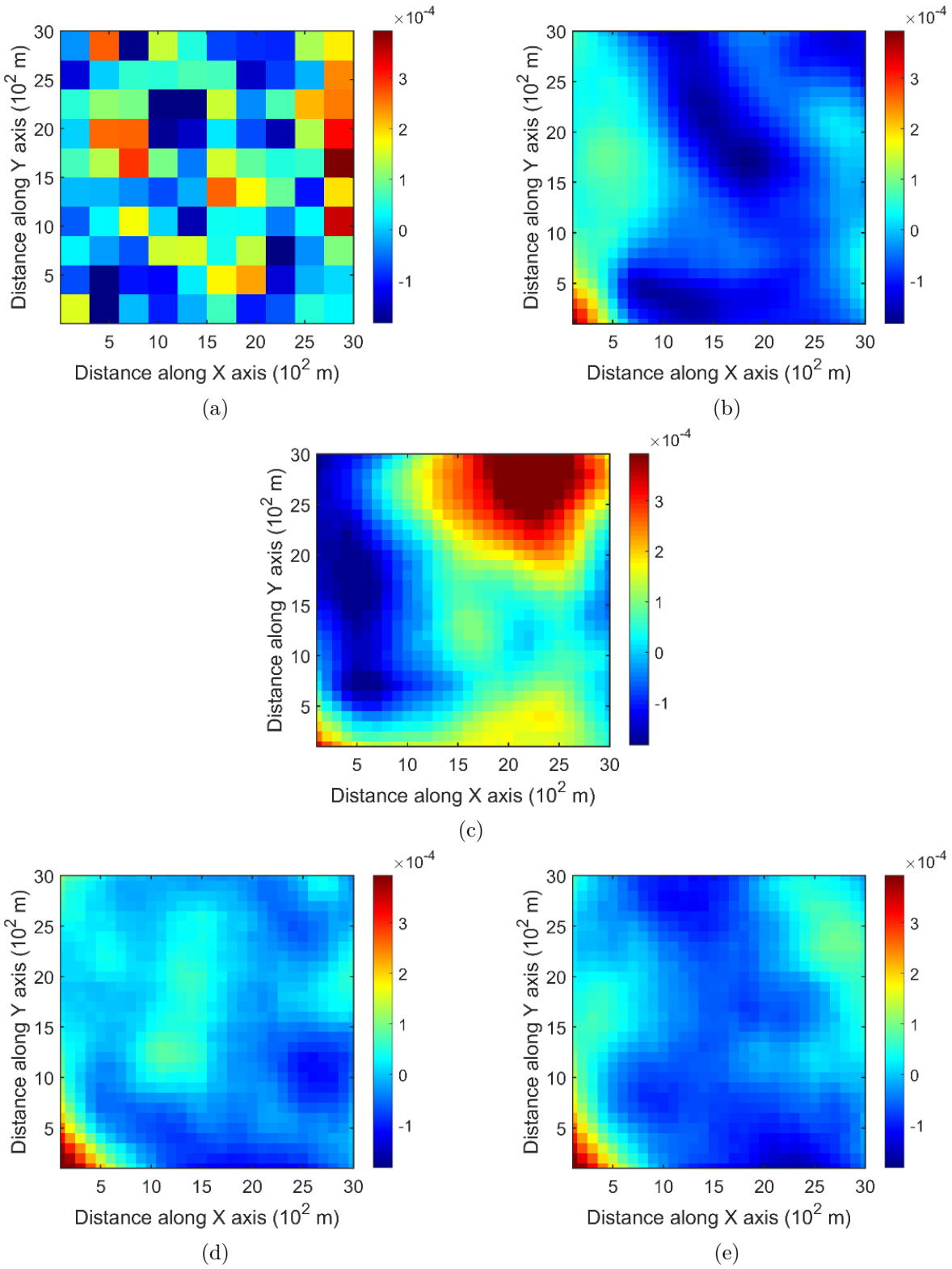


Figure 7.5 Third-order cumulant maps of a sample data; b exhaustive image; c TI; d realization in Fig. 7.2a; e realization in Fig. 7.2b

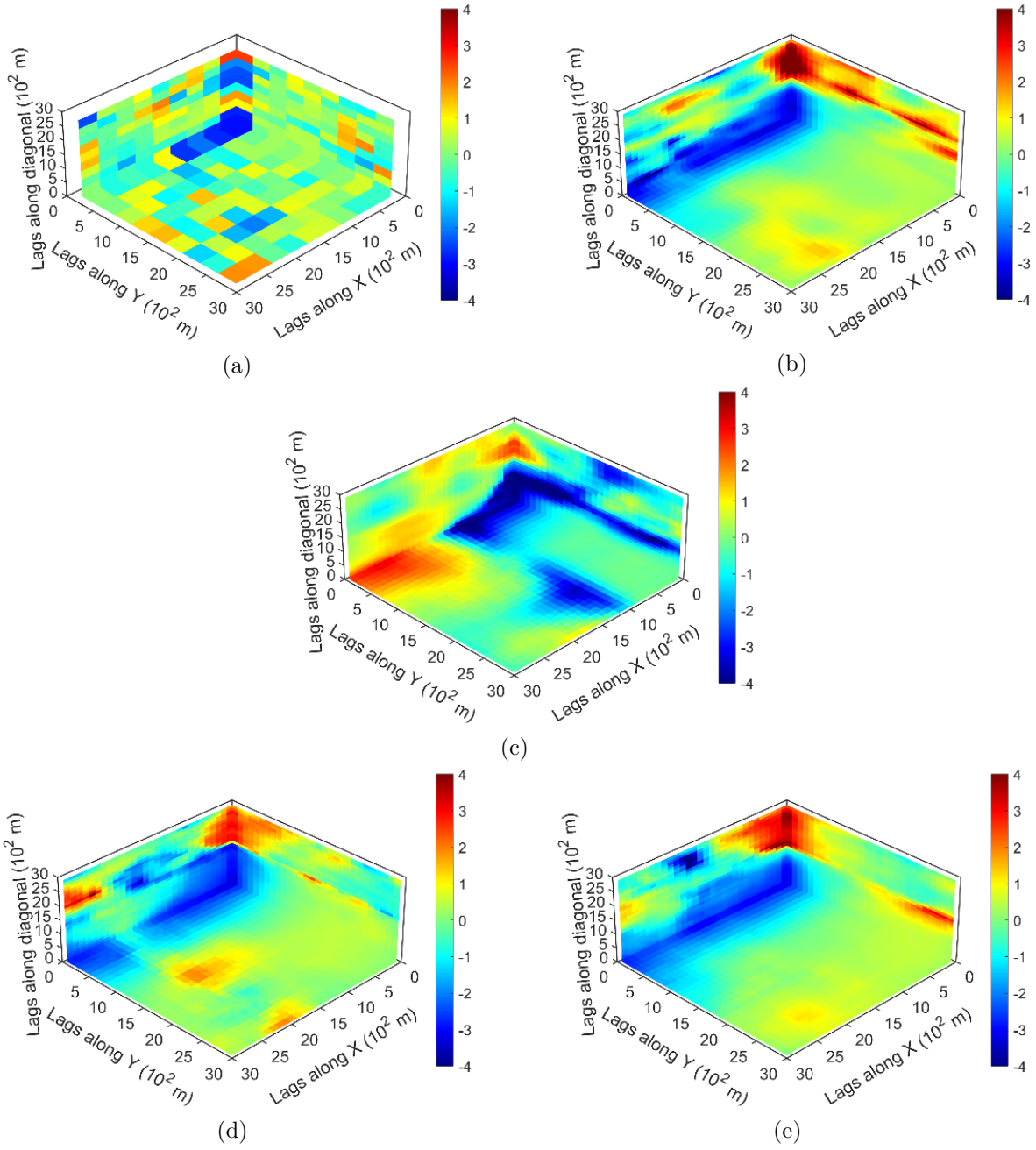
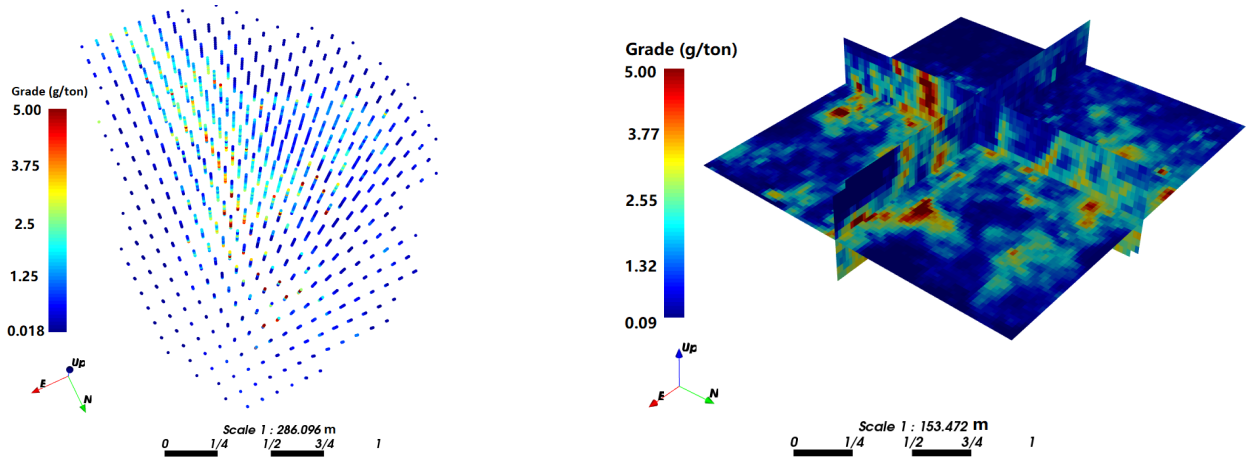


Figure 7.6 Fourth-order cumulant maps of a sample data; b exhaustive image; c TI; d realization in Fig. 7.2a; e realization in Fig. 7.2b



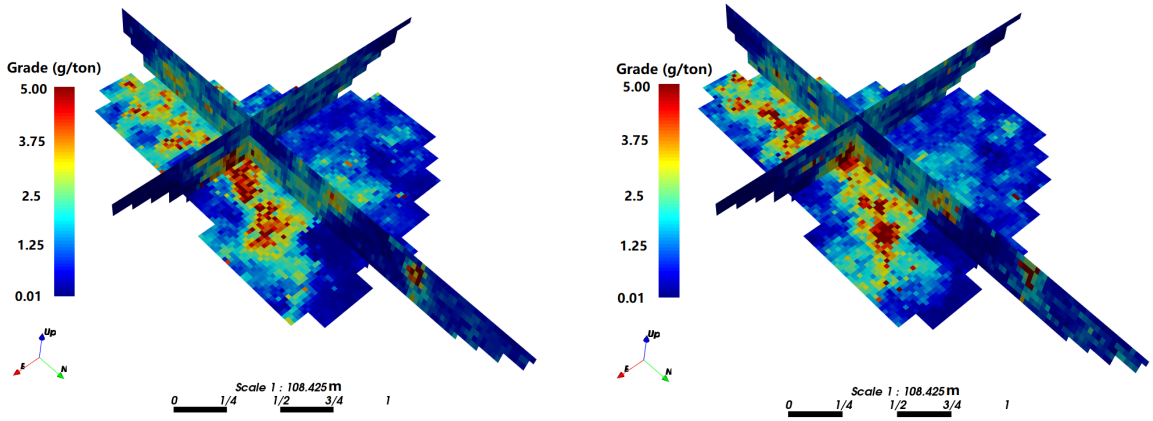
(a) (b)

Figure 7.7 a Drill hole samples at a gold deposit; b TI derived from the blast hole data in an adjacent area

Cross-sections of two different realizations are shown in Fig. 7.8. The histograms of 10 different realizations are shown in Fig. 7.9 and the comparison shows that the simulation method reproduces the histogram of the Au grades from the drill hole samples. The variograms of the same set of 10 realizations are shown in Fig. 7.10. The comparison results also show that the variograms of the simulated realizations resemble more closely the variograms of the sample data, instead of those of the TI. Fig. 7.11 shows the comparison of third-order cumulant maps of the two realizations displayed in Fig. 7.8 to the third-order cumulant maps of the sample data and the TI. The fourth-order cumulant maps are compared in the same manner and are shown in Fig. 7.12. Both the third-order or the fourth-order cumulant maps demonstrate distinct patterns compatible with the corresponding cumulant maps of the sample data. The high-order spatial information from the TI is only partly incorporated to complement the fine spatial structures of the stochastic orebody models generated with the proposed simulation method. Therefore, the high-order spatial statistics from the simulated realizations retain the main features from the sample data, reducing the influence of the possible statistical conflicts from the TI.

7.5 Conclusions

The present paper presents an extension of the high-order simulation method based on the statistical learning framework [148]. A modified concept of aggregated kernel statistics is proposed to incorporate the high-order spatial information at two different scales from the sample data and the TI. Specifically, the aggregated kernel statistics proposed herein contain



(a) (b)

Figure 7.8 Two simulated simulations using the sample data and the TI shown in Fig. 7.7

the high-order spatial information at the coarse scales from the sample data with high-order spatial information at the finer scales complemented by the TI. These aggregated kernel statistics are utilized in a kernelized learning algorithm to develop the high-order simulation method, which incorporates high-order spatial statistics from both the sample data and the TI. Although the present study only considered the data at two different scales, the proposed aggregated kernel statistics can be easily extended to scales of more than two, given that the resolutions of data sets at different scales progressively increase. In practice, it is suitable for applications where data are progressively expanding along certain time periods. A high-order simulation program based on the above paradigm is developed and described. The simulation program is integrated into the SGeMS platform for a user-friendly parameter selection and visualization in three-dimensional space. This simulation program is utilized here to carry out two different case studies. The first case study with the synthetic data set demonstrates the capacity of the proposed simulation method in reproducing the low- and high-order spatial statistics from the sample data, while significantly mitigating the statistical conflicts between the samples and the TI. The study using a gold deposit shows the practical aspects of applying the simulation program to simulate pertinent properties of actual mineral deposits.

7.6 Computer Code Availability

- Name of code: kernelsim
- Developer: Lingqing Yao
- Contact details: COSMO – Stochastic Mine Planning Laboratory, Department of Min-

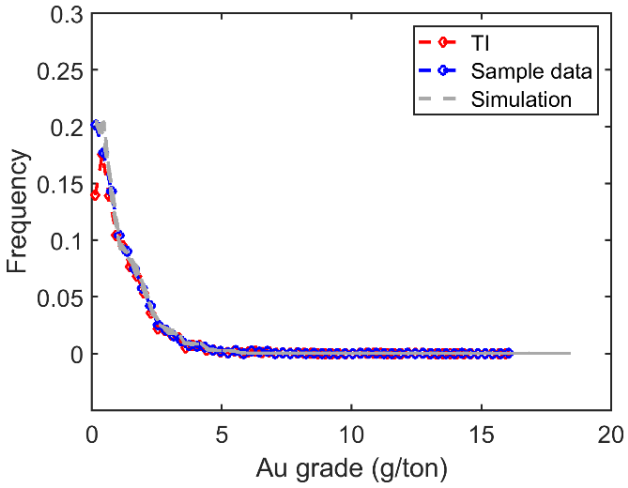


Figure 7.9 Histograms of 10 simulated realizations using the samples and the TI shown in Fig. 7.7

ing and Materials Engineering, McGill University, 3450 University Street, Montreal, QC H3A 2A7, Canada

- E-mail: yaolingqing@gmail.com

- Year first available: 2020
- Hardware required: run on a computer with 4 cores (2.4 GHz each) and 8 GB.
- Software required: Needs SGeMS software
- Program language: C ++
- Program size: 122 kb
- Details on how to access the source code: the source files of kernelsim can be downloaded from github: `https://github.com/yaolq/kernelsim`

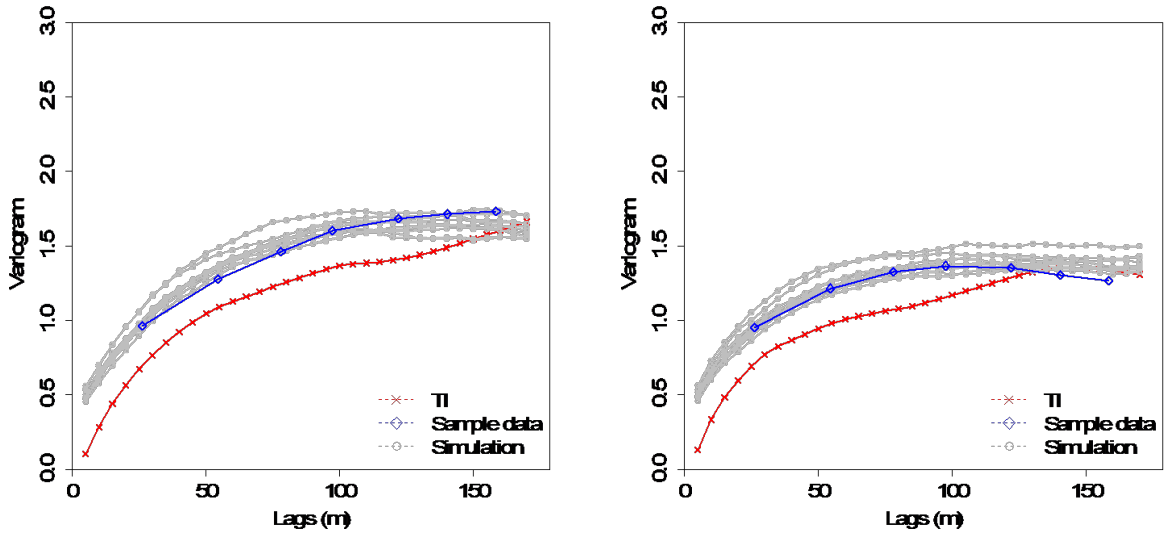


Figure 7.10 Variograms of 10 simulated realizations along a E-W and b N-S direction, using the samples and the TI shown in Fig. 7.7

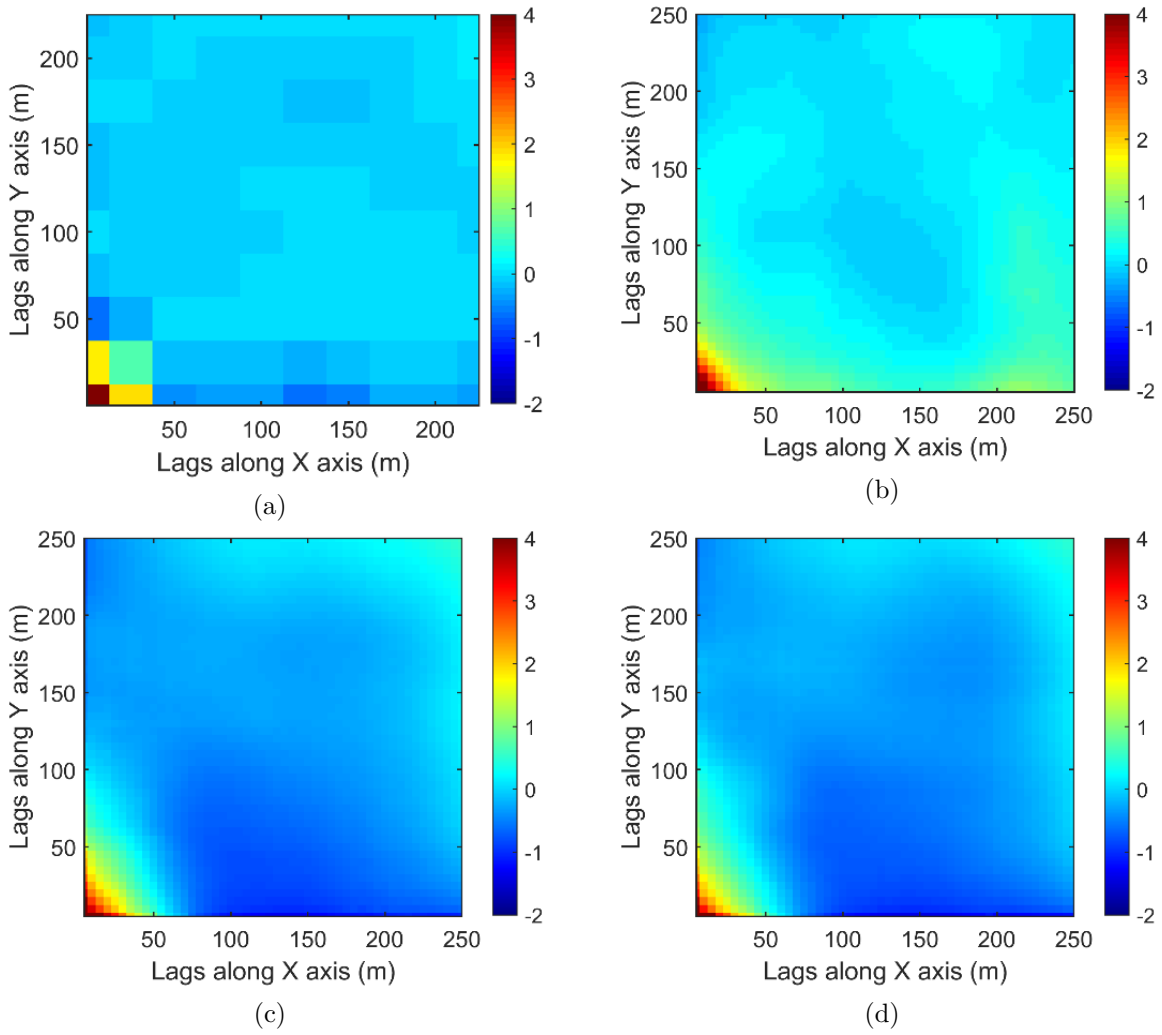


Figure 7.11 Third-order cumulant maps of a sample data; b TI; d realization in Fig. 7.8a; e realization in Fig. 7.8b

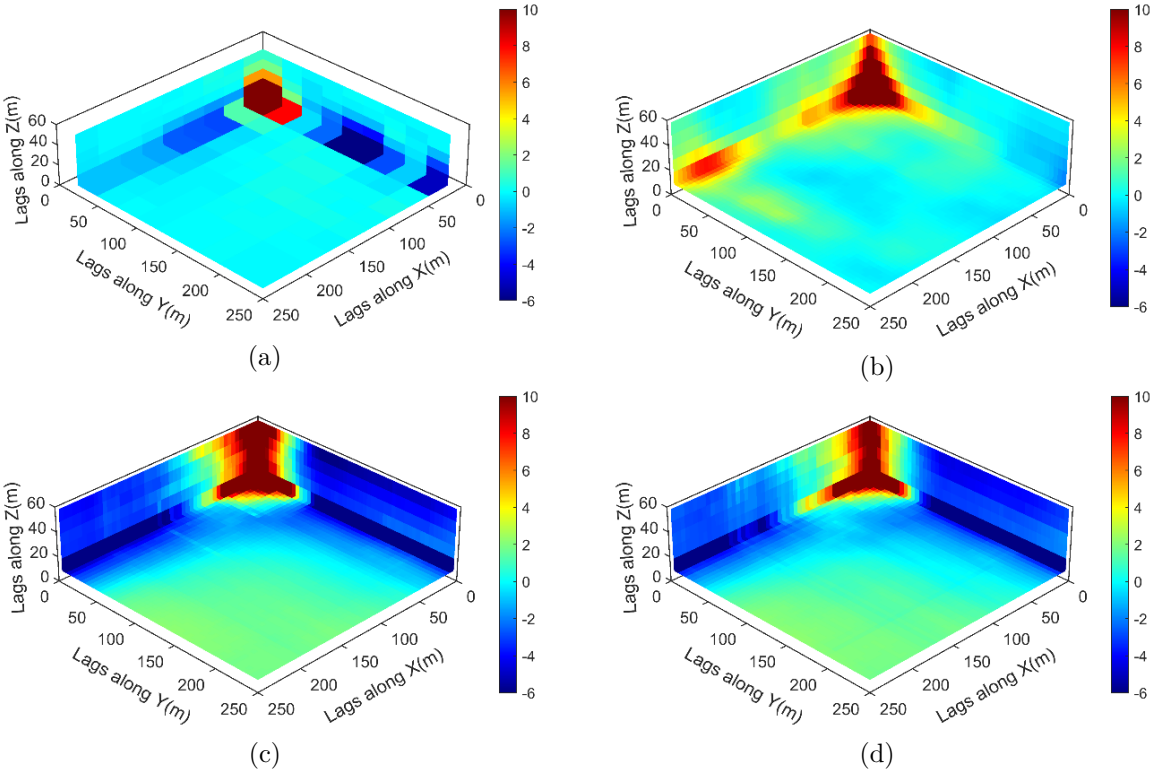


Figure 7.12 Fourth-order cumulant maps of a sample data; b TI; d realization in Fig. 7.8a; e realization in Fig. 7.8b

CHAPTER 8 GENERAL DISCUSSION

Apart from the method developments in the aforementioned chapters, this chapter stresses certain more general implications from the proposed statistical learning framework for high-order stochastic simulation.

The early development from Parzen [112] proposes probability density estimation in the kernel-like form as

$$f_n(z) = \frac{1}{n}\sum_{i=1}^{n} K_h(z - z_i), \tag{8.1}$$

where K_h corresponds to the so-called Parzen kernel, z_i are the samples and the parameter h is called the window width. The derivation of the new computational model for high-order simulation in Chapter 4 resembles the above kernel-like form (ref. Equation (4.18)), while the kernel is defined as the spatial Legendre moment kernel (see Equation (5.6)). Notable is that the derivation of the spatial Legendre moment kernel is general and thus can be easily extended to develop kernels based on other type of orthogonal bases.

Interestingly, the second-order covariance function is also positive definite and therefore determines a kernel. The covariance kernel defines a kernel Hilbert space as the dual of the random field model as discussed in Wahba [117]. Furthermore, the minimum variance linear unbiased estimator of a spatial attribute given a set of conditioning data can be written as a dual expression in the Hilbert space with the related covariance kernel [117], which is called dual kriging in terms of geostatistical theory [26]. In this sense, the spatial Legendre moment kernel proposed in this thesis has clearer meaning of defining a kernel space which carries higher-order spatial statistical information than the covariance kernel space.

From the machine learning point of view, the proposed statistical framework for high-order simulation can be categorized as generative model [151] since the probability distributions are learned from the available data. Typically the generative models assume the underlying distribution over the data has a specific parametric form, and parameters of interest are estimated from the data, such as the maximum likelihood and other methods [152]. The proposed statistical framework herein, however, does not assume any specific form of probability distribution and thus provides a non-parametric generative model for developing the high-order simulation methods. It is obvious to appreciate the difference between the proposed statistical framework and the traditional second-order stochastic simulation based on Gaussian random field models by noticing the non-parametric feature of the former. However, the more fundamental element of the present developments in this thesis is an active learning

process to minimize the gap between the observed high-order spatial statistics and the target random field model. As shown in Figure 8.1, the high-order spatial information is extracted and observed from the available data, and the target probability distribution is selected from a given solution space by the statistical learning process. Although the previous high-order simulation methods based on the approximation of probability distribution by orthogonal polynomial series are also non-parametric, a distinction is that the learning mechanism plays more significant role in the simulation framework presented in this thesis. As a result, the solution space of the target probability distribution is relatively independent from the observation space of high-order spatial statistics. As an example to show the above-mentioned distinction, the high-order simulation method based on Legendre polynomial expansion series directly use the high-order spatial cumulants to compute the coefficients of the expansion series. It means that the approximation of the probability distribution is only optimal when the solution space of the target distribution lies in the vector space of Legendre polynomials. On the contrary, the statistical learning framework offers the flexibility to choose the observation space of high-order spatial statistics and the solution space of target distributions. For example, the high-order spatial statistics can be derived from other type of polynomials than Legendre polynomials while remain the solution space unchanged as the convex space of certain prototype distributions. From this perspective, the proposed statistical framework for high-order stochastic simulation is quite general and it is possible to accommodate the new developments of methods by choosing different definitions of high-order spatial statistics or solution spaces for the random field models.

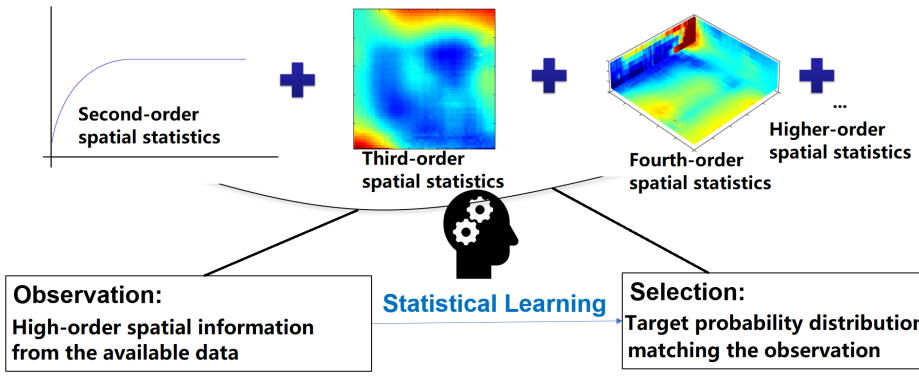


Figure 8.1 General learning statistical framework for high-order stochastic simulation

CHAPTER 9 CONCLUSION AND RECOMMENDATIONS

The present thesis emphasizes on developing new high-order sequential simulation methods driven by a learning process in kernel space. Along with development of the proposed methods, there are certain related new concepts, as well as the new statistical learning framework presented in this thesis. In the following sections, the major contributions of this thesis are summarized to highlight the advancement of knowledge regarding the high-order simulation methods. Limitations of the current research is also discussed to indicate potential directions of improvement. The future works are presented in the end.

9.1 Summary of Contributions

As proposed in Mustapha and Dimitrakopoulos [18], the original high-order simulation method takes no assumption of specific probability distribution on the random field models and the Legendre polynomial expansion series are used for approximating the probability density functions encountered in the simulation. However, it requires an explicit expansion of the Legendre polynomial series and an exhaustive computation of spatial cumulants over different orders. The high-order spatial cumulants have to be stored in a tree structure in memory, adding to the complexity of implementation. In Chapter 4, a new computational model is proposed to incorporate the spatial statistics of different orders given a spatial template into a unified function, thus avoiding the memory storage and the explicit computation of high-order spatial statistics and gaining higher computational efficiency with simplified recursive algorithms. More importantly, it turns out that the proposed computational model can be written in a kernel-like form.

Thus, a new kernel function is proposed in Chapter 5 to construct the so-called spatial Legendre moment kernel space. The high-order spatial statistics from the available data are encapsulated in the kernel statistics through the feature mapping. The target probability distributions of the random field model are also embedded into the kernel space through the concept of expected kernel statistics. Furthermore, a new statistical learning framework for high-order sequential simulation is proposed and leads to a kernelized learning algorithm, which matches the high-order spatial statistics of the target distributions to those of the available data. The learning algorithm towards high-order simulation has the generalization capacity, and thus addresses the numerical instability regarding the positiveness by approximating the probability density function via polynomial series. Notable is that the proposed statistical learning framework is general and provides new perspectives looking into the high-

order simulation. Under this framework, the high-order simulation methods are characterized by the kernelization. The main step is to design pertinent kernel functions which entail high-order spatial information of the available data, and thereafter a kernelized learning algorithm being developed to achieve the optimal target distribution for generating the realizations.

The concept of aggregated kernel statistics is proposed in Chapter 6 to efficiently utilize the incomplete high-order spatial information from the sample data. As a result, the high-order spatial statistics from the replicates with different spatial configurations are combined together through the aggregated kernel statistics. The aggregated statistics is incorporated to the statistical framework, enabling a sparse data learning algorithm and leading to a training-image free high-order simulation method. Chapter 7 proposes another type of aggregated kernel statistics which incorporates the high-order spatial information at multiple scales from the sample data and the TI. A high-order sequential simulation algorithm is developed based on learning the high-order spatial information at multiple scales through the aggregated kernel statistics. An implementation of the simulation method in C++ language is presented and the related software compatible to SGeMS platform is developed and tested with practical applications.

Various case studies are conducted to test the performances of the above-mentioned simulation methods. The results show that all the proposed methods reproduce both the low-order and high-order spatial statistics of the sample data, whereas the simulation methods based on the statistical learning framework demonstrate better stability to mitigate the possible statistical conflicts between the sample data and the TI. In terms of addressing the statistical conflicts between the samples and the TI, several different attempts are approached from different perspectives. In Chapter 5, it mainly relies on the generalization capacity of the proposed statistical learning framework, so that the high-order spatial information from the TI complies to the sample data and, thus mitigates the possible statistical conflicts. In Chapter 6, the influence of the TI is eliminated through the sparse data learning method based on the high-order spatial information from the sample data. In Chapter 7, the same goal is achieved by the concept of learning high-order spatial statistics at multiple scales. The high-order spatial statistics at coarse scales are incorporated into the simulation while the finer spatial structures are enhanced by the high-order spatial information at fine scales from the TI. The case studies at a gold deposit demonstrate the practical aspects of the proposed simulation methods in stochastic orebody modeling, as a support of mine planning decisions under uncertainty.

9.2 Limitations

The main limitations of the high-order simulation methods developed in this thesis, are related to the computational aspects in large-scale simulation, as a common problem in most of the kernel methods. The reason is that in general, the operations with kernels are applied to each sample, resulting in a data-dependant computation. Specifically in the present research, the computation of the kernel statistics is carried out by averaging the features in the kernel space mapped from each replicate retrieved from the sample data or the TI. Although the computation related to each replicate is elementary, the total computational cost depends on the number of replicates and, in turn, the size of the available data set. Retrieving the replicates from the sample data is the other factor that influences the computational cost, because the sample data are always distributed irregularly in space and constraints such as the tolerances of the lags and the angles along the spatial templates have to be considered.

9.3 Future Research

The computational complexity of high-order simulation methods developed in this thesis mainly depends on the computational efficiency of kernel statistics. As pointed out in the limitation, the computation cost of kernel statistics increases as the size of the data set increases. Nevertheless, the related operations on each replicate is elementary and remain the same regardless of the spatial locations. Thus, it is suitable to parallelize the computation of kernel statistics by distributing the elementary operations on the replicates concurrently. It's also possible to further parallelize the sequential simulation process by making subdivisions on the simulation domains. Future research on prallelization of the simulation algorithms could promisingly address the limitation of conducting the large-scale simulation based on the proposed simulation methods.

Currently, this thesis focuses on high-order simulation of continuous attributes, such as the metal grades at a mineral deposit. In mining practice, certain geological attributes, such as lithologies, are described as categorical data. Since the core method of the proposed statistical learning framework is the kernelization. Therefore, a future research could be considered to design kernels carrying high-order spatial statistics from the categorical data, and to develop the relevant high-order simulation methods for categorical data.

Moreover, because the natural attributes always coexist and interact with each other in space. In addition to the spatial correlations of each single attribute of interest, it is also important to model the cross-interrelations among all the attributes to better represent the spatial uncertainty. Therefore, an interesting research is to design kernels working with

high-order cross-interrelations, and to further develop the high-order simulation methods for multi-variate attributes.

Lastly, the present research clearly defines a learning mechanism for developing the high-order simulation methods. The kernel statistics of the available data represent the high-order spatial information for the target random field model to learn. It is possible to combine the deep learning techniques for training the kernel statistics from the available data and making predictions for the unknown situations. The combination of kernel methods with deep learning avoids the repeating computation of kernel statistics during the sequential simulation process, by using the pre-trained model to make predictions. It is promising to take research along this line to handle the learning from a large data set, and to initiate online updating with incoming data provided.

REFERENCES

[1] M. Vallee, *Mineral resource + engineering, economic and legal feasibility = ore reserve.* Montreal, QC, Canada: Canadian Institute of Mining, Metallurgy and Petroleum, 2000, vol. 93.

[2] P. Ravenscroft, "Risk analysis for mine scheduling by conditional simulation," *Transactions of the Institution of Mining and Metallurgy. Section A. Mining Industry*, vol. 101, pp. 104–108, 1992.

[3] P. Dowd, "Risk assessment in reserve estimation and open-pit planning," *Transactions of the Institution of Mining and Metallurgy(Section A: Mining Industry)*, vol. 103, pp. 148–154, 1994.

[4] R. Dimitrakopoulos, C. T. Farrelly, and M. Godoy, "Moving forward from traditional optimization: grade uncertainty and risk effects in open-pit design," *Mining Technology*, vol. 111, no. 1, pp. 82–88, 2002.

[5] R. Dimitrakopoulos and S. Ramazan, "Stochastic integer programming for optimising long term production schedules of open pit mines: methods, application and value of stochastic solutions," *Mining Technology*, vol. 117, no. 4, pp. 155–160, 2008.

[6] J. R. Birge and F. Louveaux, *Two-Stage recourse problems.* New York, NY: Springer New York, 2011, pp. 181–263.

[7] S. Ramazan and R. Dimitrakopoulos, "Production scheduling with uncertain supply: a new solution to the open pit mining problem," *Optimization and Engineering*, vol. 14, no. 2, pp. 361–380, 2013.

[8] R. Goodfellow and R. Dimitrakopoulos, "Global optimization of open pit mining complexes with uncertainty," *Applied Soft Computing*, vol. 40, pp. 292–304, 2016.

[9] ——, "Simultaneous stochastic optimization of mining complexes and mineral value chains," *Mathematical Geosciences*, vol. 49, no. 3, pp. 341–360, 2017.

[10] P. Ravenscroft, *Conditional simulation for mining: practical implementation in an industrial environment.* Dordrecht: Springer Netherlands, 1994, pp. 79–87.

[11] S. De Iaco and S. Maggio, "Validation techniques for geological patterns simulations based on variogram and multiple-point statistics," *Mathematical Geosciences*, vol. 43, no. 4, pp. 483–500, 2011.

[12] F. Guardiano and R. M. Srivastava, *Multivariate geostatistics: beyond bivariate moments*, ser. Quantitative Geology and Geostatistics. Kluwer Academic, Dordrecht, 1993, vol. 5, book section 12, pp. 133–144.

[13] A. G. Journel, *Deterministic geostatistics: A new visit.* Dordrecht: Kluwer Academic, 1997, vol. 1, pp. 292–301.

[14] W. Xu, "Conditional curvilinear stochastic simulation using pixel-based algorithms," *Mathematical Geology*, vol. 28, no. 7, pp. 937–949, 1996.

[15] A. G. Journel, "Geostatistics for conditional simulation of ore bodies," *Economic Geology*, vol. 69, no. 5, pp. 673–687, 1974.

[16] T. M. Cover and J. A. Thomas, *Elements of information theory.* Hoboken, N.J.: Wiley-Interscience, 2006.

[17] R. Dimitrakopoulos, H. Mustapha, and E. Gloaguen, "High-order statistics of spatial random fields: exploring spatial cumulants for modeling complex non-gaussian and non-linear phenomena," *Mathematical Geosciences*, vol. 42, no. 1, pp. 65–99, 2010.

[18] H. Mustapha and R. Dimitrakopoulos, "High-order stochastic simulation of complex spatially distributed natural phenomena," *Mathematical Geosciences*, vol. 42, no. 5, pp. 457–485, 2010.

[19] ——, "Hosim: A high-order stochastic simulation algorithm for generating three-dimensional complex geological patterns," *Computers & Geosciences*, vol. 37, no. 9, pp. 1242–1253, 2011.

[20] G. Matheron, "The intrinsic random functions and their applications," *Advances in Applied Probability*, vol. 5, no. 3, pp. 439–468, 1973.

[21] M. W. Davis, "Production of conditional simulations via the lu triangular decomposition of the covariance-matrix," *Mathematical Geology*, vol. 19, no. 2, pp. 91–98, 1987.

[22] C. V. Deutsch and A. G. Journel, *GSLIB geostatistical software library and user's guide*, 2nd ed. New York: Oxford University Press, 1992.

[23] A. Journel, *Modeling uncertainty: Some conceptual thoughts*, ser. Quantitative Geology and Geostatistics. Springer, Dordrecht, 1994, vol. 6, book section 5, pp. 30–43.

[24] M. E. Johnson, *Multivariate generation techniques.* John Wiley & Sons, Inc., 1987, pp. 43–48.

[25] M. Rosenblatt, "Remarks on a multivariate transformation," *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 470–472, 1952.

[26] P. Goovaerts, *Geostatistics for natural resources evaluation*, ser. Applied geostatistics. Oxford, England: Oxford University Press, 1997.

[27] G. Verly, *Sequential Gaussian simulation: A Monte Carlo method for generating models of porosity and permeability.* Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 345–356.

[28] R. Dimitrakopoulos and X. Luo, "Generalized sequential gaussian simulation on group size ν and screen-effect approximations for large field simulations," *Mathematical Geology*, vol. 36, no. 5, pp. 567–591, 2004.

[29] J. A. Vargas-Guzmán and R. Dimitrakopoulos, "Conditional simulation of random fields by successive residuals," *Mathematical Geology*, vol. 34, no. 5, pp. 597–611, 2002.

[30] J. A. Vargas-Guzmán and T.-C. Jim Yeh, "Sequential kriging and cokriging: Two powerful geostatistical approaches," *Stochastic Environmental Research and Risk Assessment*, vol. 13, no. 6, pp. 416–435, 1999.

[31] A. Jewbali and R. Dimitrakopoulos, "Implementation of conditional simulation by successive residuals," *Computers & Geosciences*, vol. 37, no. 2, pp. 129–142, 2011.

[32] C. V. Deutsch, "A sequential indicator simulation program for categorical variables with point and block data: Blocksis," *Computers & Geosciences*, vol. 32, no. 10, pp. 1669–1681, 2006.

[33] P. Goovaerts, "Comparative performance of indicator algorithms for modeling conditional probability distribution functions," *Mathematical Geology*, vol. 26, no. 3, pp. 389–411, 1994.

[34] A. G. Journel and F. Alabert, "Non-gaussian data expansion in the earth sciences," *Terra Nova*, vol. 1, no. 2, pp. 123–134, 1989.

[35] A. G. Journel and E. H. Isaaks, "Conditional indicator simulation: Application to a Saskatchewan uranium deposit," *Mathematical Geology*, vol. 16, no. 7, pp. 685–718, 1984.

[36] A. G. Journel, "Nonparametric estimation of spatial distributions," *Mathematical Geology*, vol. 15, no. 3, pp. 445–468, 1983.

[37] H. Wackernagel, P. Petitgas, and Y. Touffait, *Overview of methods for coregionalization analysis*. Dordrecht: Springer Netherlands, 1989, pp. 409–420.

[38] M. Goulard and M. Voltz, "Linear coregionalization model: Tools for estimation and choice of cross-variogram matrix," *Mathematical Geology*, vol. 24, no. 3, pp. 269–286, 1992.

[39] A. G. Journel and C. J. Huijbregts, *Mining geostatistics*. London; New York: Academic Press, 1978.

[40] M. David, *Handbook of applied advanced geostatistical ore reserve estimation*. Amsterdam: Elsevier, 1988.

[41] B. M. Davis and K. A. Greenes, "Estimation using spatially distributed multivariate data: An example with coal quality," *Mathematical Geology*, vol. 15, no. 2, pp. 287–300, 1983.

[42] P. Goovaerts, "Spatial orthogonality of the principal components computed from coregionalized variables," *Mathematical Geology*, vol. 25, no. 3, pp. 281–302, 1993.

[43] P. Switzer and A. A. Green, "Min/max autocorrelation factors for multivariate spatial imagery," *Computer science and statistics*, pp. 13–16, 1984.

[44] A. J. Desbarats and R. Dimitrakopoulos, "Geostatistical simulation of regionalized pore-size distributions using min/max autocorrelation factors," *Mathematical Geology*, vol. 32, no. 8, pp. 919–942, 2000.

[45] O. Rondon, "Teaching aid: Minimum/maximum autocorrelation factors for joint simulation of attributes," *Mathematical Geosciences*, vol. 44, no. 4, pp. 469–504, 2012.

[46] D. Marcotte, *Direct conditional simulation of block grades*. Dordrecht: Springer Netherlands, 1994, pp. 245–252.

[47] X. Emery, "Change of support for estimating local block grade distributions," *Mathematical Geosciences*, vol. 40, no. 6, pp. 671–688, 2008.

[48] ——, "Change-of-support models and computer programs for direct block-support simulation," *Computers & Geosciences*, vol. 35, no. 10, pp. 2047–2056, 2009.

[49] M. Godoy, "The effective management of geological risk in long-term scheduling of open pit mines," PhD Thesis, University of Queensland, 2002.

[50] J. Benndorf and R. Dimitrakopoulos, "New efficient methods for conditional simulation of large orebodies," *Orebody modelling and strategic mine planning, Spectrum Series*, vol. 14, pp. 61–67, 2007.

[51] A. Boucher and R. Dimitrakopoulos, "Multivariate block-support simulation of the yandi iron ore deposit, western australia," *Mathematical Geosciences*, vol. 44, no. 4, pp. 449–468, 2012.

[52] ——, "Block simulation of multiple correlated variables," *Mathematical Geosciences*, vol. 41, no. 2, pp. 215–237, 2009.

[53] A. Journel, "Roadblocks to the evaluation of ore reserves—the simulation overpass and putting more geology into numerical models of deposits," *Orebody modeling and strategic mine planning, AusIIMM, Melbourn, 2nd Edition, Spectrum Series*, vol. 14, pp. 29–32, 2007.

[54] S. Krishnan and A. G. Journel, "Spatial connectivity: from variograms to multiple-point measures," *Mathematical Geology*, vol. 35, no. 8, pp. 915–925, 2003.

[55] A. G. Journel and T. Zhang, "The necessity of a multiple-point prior model," *Mathematical Geology*, vol. 38, no. 5, pp. 591–610, 2006.

[56] G. Mariethoz and J. Caers, *Multiple-point geostatistics: stochastic modeling with training images*. Hoboken: Wiley, 2014.

[57] S. Strebelle, "Conditional simulation of complex geological structures using multiple-point statistics," *Mathematical Geology*, vol. 34, no. 1, pp. 1–21, 2002.

[58] J. Straubhaar *et al.*, "An improved parallel multiple-point algorithm using a list approach," *Mathematical Geosciences*, vol. 43, no. 3, pp. 305–328, 2011.

[59] J. Straubhaar, A. Walgenwitz, and P. Renard, "Parallel multiple-point statistics algorithm based on list and tree structures," *Mathematical Geosciences*, vol. 45, no. 2, pp. 131–147, 2013.

[60] G. Mariethoz, P. Renard, and J. Straubhaar, "The direct sampling method to perform multiple-point geostatistical simulations," *Water Resources Research*, vol. 46, no. 11, p. W11536, 2010.

[61] E. Meerschman *et al.*, "A practical guide to performing multiple-point statistical simulations with the direct sampling algorithm," *Computers & Geosciences*, vol. 52, pp. 307–324, 2013.

[62] S. Strebelle and C. Cavelius, "Solving speed and memory issues in multiple-point statistics simulation program snesim," *Mathematical Geosciences*, vol. 46, no. 2, pp. 171–186, 2014.

[63] T. Huang *et al.*, "Gpu-based snesim implementation for multiple-point statistical simulation," *Computers & Geosciences*, vol. 54, pp. 75–87, 2013.

[64] ——, "Gpu-accelerated direct sampling method for multiple-point statistical simulation," *Computers & Geosciences*, vol. 57, pp. 13–23, 2013.

[65] G. B. Arpat, "Sequential simulation with patterns," PhD Thesis, Stanford University, 2005.

[66] T. Zhang, P. Switzer, and A. Journel, "Filter-based classification of training image patterns for spatial simulation," *Mathematical Geology*, vol. 38, no. 1, pp. 63–80, 2006.

[67] M. Honarkhah and J. Caers, "Stochastic simulation of patterns using distance-based pattern modeling," *Mathematical Geosciences*, vol. 42, no. 5, pp. 487–517, 2010.

[68] H. Maitre *et al.*, "Feature selection for satellite image indexing," *ESA-EUSC: image information mining, Frascati, Italy*, 2005.

[69] S. Chatterjee, R. Dimitrakopoulos, and H. Mustapha, "Dimensional reduction of pattern-based simulation using wavelet analysis," *Mathematical Geosciences*, vol. 44, no. 3, pp. 343–374, 2012.

[70] S. Chatterjee, H. Mustapha, and R. Dimitrakopoulos, "Fast wavelet-based stochastic simulation using training images," *Computational Geosciences*, vol. 20, no. 3, pp. 399–420, 2016.

[71] H. Mustapha, S. Chatterjee, and R. Dimitrakopoulos, "Cdfsim: Efficient stochastic simulation through decomposition of cumulative distribution functions of transformed spatial patterns," *Mathematical Geosciences*, vol. 46, no. 1, pp. 95–123, 2014.

[72] H. Rezaee *et al.*, "Multiple-point geostatistical simulation using the bunch-pasting direct sampling method," *Computers & Geosciences*, vol. 54, no. 0, pp. 293–308, 2013.

[73] C. Gardet, M. Le Ravalec, and E. Gloaguen, "Pattern-based conditional simulation with a raster path: a few techniques to make it more efficient," *Stochastic Environmental Research and Risk Assessment*, vol. 30, no. 2, pp. 429–446, 2016.

[74] K. Mahmud *et al.*, "Simulation of earth textures by conditional image quilting," *Water Resources Research*, vol. 50, no. 4, pp. 3088–3107, 2014.

[75] A. Parra and J. M. Ortiz, "Adapting a texture synthesis algorithm for conditional multiple point geostatistical simulation," *Stochastic Environmental Research and Risk Assessment*, vol. 25, no. 8, pp. 1101–1111, 2011.

[76] L.-Y. Wei, "Texture synthesis by fixed neighborhood searching," PhD Thesis, Stanford University, 2002.

[77] P. Tahmasebi, A. Hezarkhani, and M. Sahimi, "Multiple-point geostatistical modeling based on the cross-correlation functions," *Computational Geosciences*, vol. 16, no. 3, pp. 779–797, 2012.

[78] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, Conference Proceedings, pp. 341–346.

[79] G. Mariethoz and S. Lefebvre, "Bridges between multiple-point geostatistics and texture synthesis: Review and guidelines for future research," *Computers & Geosciences*, vol. 66, no. 0, pp. 66–80, 2014.

[80] T. T. Tran, "Improving variogram reproduction on dense simulation grids," *Computers & Geosciences*, vol. 20, no. 7, pp. 1161–1168, 1994.

[81] S. N. Chiu *et al.*, *Stochastic geometry and its applications*. John Wiley & Sons, 2013.

[82] C. V. Deutsch and T. T. Tran, "Fluvsim: a program for object-based stochastic modeling of fluvial depositional systems," *Computers & Geosciences*, vol. 28, no. 4, pp. 525–535, 2002.

[83] H. H. Haldorsen and L. W. Lake, "A new approach to shale management in field-scale models," *Society of Petroleum Engineers Journal*, vol. 24, no. 04, pp. 447–457, 1984.

[84] A. G. Journel, "Multiple-point geostatistics: A state of the art," Stanford, Report no. 16, Stanford Center for Reservoir Forecasting, 2003.

[85] A. Maharaja, "Tigenerator: Object-based training image generator," *Computers & Geosciences*, vol. 34, no. 12, pp. 1753–1761, 2008.

[86] M. G. Kendall, A. Stuart, and J. K. Ord, *Kendall's advanced theory of statistics.* New York: Oxford University Press, 1987.

[87] C. L. Nikias and A. P. Petropulu, *Higher-order spectra analysis : a nonlinear signal processing framework.* Englewood Cliffs, N.J.: PTR Prentice Hall, 1993.

[88] T. P. Speed, "Cumulants and partition lattices," *Australian Journal of Statistics*, vol. 25, no. 2, pp. 378–388, 1983.

[89] P. J. Smith, "A recursive formulation of the old problem of obtaining moments from cumulants and vice versa," *The American Statistician*, vol. 49, no. 2, pp. 217–218, 1995.

[90] S. De Iaco, "On the use of different metrics for assessing complex pattern reproductions," *Journal of Applied Statistics*, vol. 40, no. 4, pp. 808–822, 2013.

[91] H. Mustapha and R. Dimitrakopoulos, "A new approach for geological pattern recognition using high-order spatial cumulants," *Computers & Geosciences*, vol. 36, no. 3, pp. 313–334, 2010.

[92] X. Li *et al.*, "Accelerating experimental high-order spatial statistics calculations using gpus," *Computers & Geosciences*, vol. 70, pp. 128–137, 2014.

[93] N. N. Lebedev and R. A. Silverman, *Special functions and their applications.* Englewood Cliffs, N.J.: Prentice-Hall, 1965.

[94] C. J. Zarowski, *Orthogonal polynomials.* John Wiley & Sons, Inc., 2004, pp. 207–250.

[95] H. Mustapha and R. Dimitrakopoulos, "Generalized laguerre expansions of multivariate probability densities with moments," *Computers & Mathematics with Applications*, vol. 60, no. 7, pp. 2178–2189, 2010.

[96] J. A. Vargas-Guzmán, "Heavy tailed probability distributions for non-gaussian simulations with higher-order cumulant parameters predicted from sample data," *Stochastic Environmental Research and Risk Assessment*, vol. 26, no. 6, pp. 765–776, 2012.

[97] A. A. H. Abolhassani, R. Dimitrakopoulos, and F. P. Ferrie, *A new high-order, nonstationary, and transformation invariant spatial simulation approach.* Cham: Springer International Publishing, 2017, pp. 93–106.

[98] I. Minniakhmetov and R. Dimitrakopoulos, *A high-order, data-driven framework for joint simulation of categorical variables.* Cham: Springer International Publishing, 2017, pp. 287–301.

[99] ——, "Joint high-order simulation of spatially correlated variables using high-order spatial statistics," *Mathematical Geosciences*, vol. 49, no. 1, pp. 39–66, 2016.

[100] J. Caers, "Stochastic simulationwith neural networks," Stanford, Report, 1998, report no. 11, Stanford Center for Reservoir Forecasting.

[101] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[102] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[103] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, Conference Proceedings, pp. 2672–2680.

[104] L. Mosser, O. Dubrule, and M. J. Blunt, "Reconstruction of three-dimensional porous media using generative adversarial neural networks," *Phys. Rev. E*, vol. 96, p. 043309, Oct 2017.

[105] E. Dupont *et al.*, "Generating realistic geology conditioned on physical measurements with generative adversarial networks," 2018.

[106] S. Chan and A. H. Elsheikh, "Parametric generation of conditional geological realizations using generative neural networks," *Computational Geosciences*, vol. 23, no. 5, pp. 925–952, 2019.

[107] L. Azevedo *et al.*, "Generative adversarial network as a stochastic subsurface model reconstruction," *Computational Geosciences*, 2020.

[108] S. Avalos and J. M. Ortiz, "Recursive convolutional neural networks in a multiple-point statistics framework," *Computers & Geosciences*, vol. 141, p. 104522, 2020.

[109] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[110] V. N. Vapnik, *Statistical learning theory*. New York: Wiley, 1998.

[111] B. Scholkopf and A. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Cambridge, Mass.: MIT Press, 2001.

[112] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[113] L. Breiman, W. Meisel, and E. Purcell, "Variable kernel estimates of multivariate densities," *Technometrics*, vol. 19, no. 2, pp. 135–144, 1977.

[114] G. R. Terrell and D. W. Scott, "Variable kernel density estimation," *The Annals of Statistics*, vol. 20, no. 3, pp. 1236–1265, 1992.

[115] D. W. Scott, *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[116] A. Berlinet and C. Thomas-Agnan, *Reproducing kernel Hilbert spaces in probability and statistics*. Boston: Kluwer Academic, 2004.

[117] G. Wahba, *Spline models for observational data*. SIAM, 1990.

[118] C. Scheidt and J. Caers, "Representing spatial uncertainty using distances and kernels," *Mathematical Geosciences*, vol. 41, no. 4, pp. 397–419, 2009.

[119] G. B. Arpat and J. Caers, "Conditional simulation with patterns," *Mathematical Geology*, vol. 39, no. 2, pp. 177–203, 2007.

[120] J. Wu, A. Boucher, and T. Zhang, "A sgems code for pattern simulation of continuous and categorical variables: Filtersim," *Computers & Geosciences*, vol. 34, no. 12, pp. 1863–1876, 2008.

[121] A. Boucher, "Considering complex training images with search tree partitioning," *Computers & Geosciences*, vol. 35, no. 6, pp. 1151–1158, 2009.

[122] N. Remy, A. Boucher, and J. Wu, *Applied geostatistics with SGeMS : a user's guide*. Cambridge, UK: Cambridge University Press, 2009.

[123] A. Boucher *et al.*, "Simulation of geological contacts from interpreted geological model using multiple-point statistics," *Mathematical Geosciences*, vol. 46, no. 5, pp. 1–12, 2014.

[124] X. Li *et al.*, "Patch-based iterative conditional geostatistical simulation using graph cuts," *Water Resources Research*, vol. 52, no. 8, pp. 6297–6320, 2016.

[125] V. Osterholt and R. Dimitrakopoulos, *Simulation of wireframes and geometric features with multiple-point techniques: application at Yandi iron ore deposit, Australia*, 2nd ed. AusIMM Spectrum Series, 2007, vol. 14, pp. 51–60.

[126] R. Goodfellow *et al.*, "Quantifying multi-element and volumetric uncertainty, coleman mccreedy deposit, ontario, canada," *Computers & Geosciences*, vol. 42, no. 0, pp. 71–78, 2012.

[127] I. Minniakhmetov and R. Dimitrakopoulos, "Joint high-order simulation of spatially correlated variables using high-order spatial statistics," *Mathematical Geosciences*, vol. 49, no. 1, pp. 39–66, 2017.

[128] I. Minniakhmetov, R. Dimitrakopoulos, and M. Godoy, "High-order spatial simulation using legendre-like orthogonal splines," *Math Geosci*, vol. 50, no. 7, pp. 753–780, 2018.

[129] S. Mao and A. Journel, "Generation of a reference petrophysical/seismic data set: the stanford v reservoir," Stanford, Report no. 12, Stanford Center for Reservoir Forecasting, 1999.

[130] M. A. Cohen and C. O. Tan, "A polynomial approximation for arbitrary functions," *Applied Mathematics Letters*, vol. 25, no. 11, pp. 1947–1952, 2012.

[131] H. Wang and S. Xiang, *On the convergence rates of Legendre approximation*, 2012, vol. 81.

[132] J. P. de Carvalho, R. Dimitrakopoulos, and I. Minniakhmetov, "High-order block support spatial simulation method and its application at a gold deposit," *Mathematical Geosciences*, vol. 51, no. 6, pp. 793–810, 2019.

[133] L. Yao, R. Dimitrakopoulos, and M. Gamache, "A new computational model of high-order stochastic simulation based on spatial legendre moments," *Mathematical Geosciences*, vol. 50, no. 8, pp. 929–960, 2018.

[134] V. N. Vapnik, *The nature of statistical learning theory.* New York: Springer, 1995.

[135] A. Smola *et al.*, *A Hilbert Space Embedding for Distributions*. Berlin, Heidelberg: Springer, 2007, pp. 13–31.

[136] L. Song *et al.*, "Tailoring density estimation via reproducing kernel moment matching," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, Conference Proceedings, pp. 992–999.

[137] L. Song, K. Fukumizu, and A. Gretton, "Kernel embeddings of conditional distributions: a unified kernel framework for nonparametric inference in graphical models," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 98–111, 2013.

[138] K. Muandet *et al.*, "Kernel mean embedding of distributions: a review and beyonds," *arXiv preprint arXiv:1605.09522*, 2016.

[139] E. M. Stein and R. Shakarchi, *Real analysis : measure theory, integration, and Hilbert spaces*. Princeton and Oxford: Princeton University Press, 2005.

[140] I. Steinwart and A. Christmann, *Support vector machines*. New York: Springer, 2008.

[141] Y. Altun and A. Smola, "Unifying divergence minimization and statistical inference via convex duality," in *Proceedings of the 19th annual conference on Learning Theory*. Springer-Verlag, 2006, Conference Proceedings, pp. 139–153.

[142] V. N. Vapnik and S. Mukherjee, "Support vector method for multivariate density estimation," in *Proceedings of the 12th International Conference on Neural Information Processing Systems*. MIT Press, 1999, Conference Proceedings, pp. 659–665.

[143] D. Goldfarb and A. Idnani, "A numerically stable dual method for solving strictly convex quadratic programs," *Mathematical Programming*, vol. 27, no. 1, pp. 1–33, 1983.

[144] S. A. Vavasis, *Complexity theory: quadratic programming*. Boston, MA: Springer, 2001, pp. 304–307.

[145] A. Journel and C. Deutsch, "Entropy and spatial disorder," *Mathematical Geology*, vol. 25, no. 3, pp. 329–355, 1993.

[146] A. G. Journel, *Beyond Covariance: The advent of multiple-point geostatistics*. Dordrecht: Springer Netherlands, 2005, pp. 225–233.

[147] S. Strébelle, "Sequential simulation drawing structures from training images," PhD thesis, Stanford University, 2000.

[148] L. Yao, R. Dimitrakopoulos, and M. Gamache, "High-order sequential simulation via statistical learning in reproducing kernel hilbert space," *Mathematical Geosciences*, 2019.

[149] ——, "Training-image-free high-order stochastic simulation based on aggregated kernel statistics," *Mathematical Geosciences*, 2020, (Submitted).

[150] N. Remy *et al.*, "Gstl: the geostatistical template library in c++," *Computers & Geosciences*, vol. 28, no. 8, pp. 971–979, 2002.

[151] T. Jebara, *Machine learning: discriminative and generative*. Springer Science & Business Media, 2012, vol. 755.

[152] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge: Cambridge University Press, 2014.

APPENDIX A EXPANSION SERIES OF PROBABILITY DENSITY FUNCTION BASED ON THE SPATIAL LEGENDRE MOMENTS

Suppose that the multivariate function $f(z_0, z_1, \ldots, z_N)$ is the density function related to the joint distribution of random variables on a spatial template $\mathbf{T}$, and that it can be expressed as a Legendre polynomial series. The sequence of Legendre polynomials at different orders forms a set of orthogonal bases of a Hilbert space containing all the continuous functions defined on $D = [-1,\ 1]^{N+1}$; the inner product is defined as

$$\langle g, h\rangle = \int_D gh\ \mathrm{d}z_0 \ldots \mathrm{d}z_N, \tag{A.1}$$

where g, h are functions in the Hilbert space.

From the orthogonal property of Legendre polynomial and the definition of its norm shown in Equation (4.7), there is:

$$f\left(z_0, z_1, \ldots, z_N\right) = \sum_{w_0=0}^{\infty}\sum_{w_1=0}^{\infty}\cdots\sum_{w_N=0}^{\infty}\langle f, \overline{P}_{w_0}\overline{P}_{w_1}\cdots\overline{P}_{w_N}\rangle\overline{P}_{w_0}\overline{P}_{w_1}\cdots\overline{P}_{w_N}, \tag{A.2}$$

where the set $\{\overline{P}_{w_0}\overline{P}_{w_1}\cdots\overline{P}_{w_N}|w_i = 0, 1, 2, \ldots,\ 0 \le i \le N\}$ are the orthonormal bases of the Hilbert space, and $\overline{P}_{w_i}\left(z_i\right) = \frac{P_{w_i}(z_i)}{\|P_{w_i}\|},\ 0 \le i \le N$, is the normalized Legendre polynomial. Therefore,

$$\overline{P}_{w_0}\overline{P}_{w_1}\cdots\overline{P}_{w_N} = \frac{P_{w_0}P_{w_1}\cdots P_{w_N}}{\|\ P_{w_0}\ \|\cdots\|\ P_{w_N}\ \|} = \prod_{i=0}^{N}\sqrt{w_i + \frac{1}{2}}\cdot P_{w_0}P_{w_1}\cdots P_{w_N} \tag{A.3}$$

$$\langle f, \overline{P}_{w_0}\overline{P}_{w_1}\cdots\overline{P}_{w_N}\rangle = \prod_{i=0}^{N}\sqrt{w_i + \frac{1}{2}}\cdot\langle f, P_{w_0}P_{w_1}\cdots P_{w_N}\rangle. \tag{A.4}$$

Combining Equations (A.2)–(A.4), it is:

$$f\left(z_0, z_1, \ldots, z_N\right) = \sum_{w_0=0}^{\infty}\sum_{w_1=0}^{\infty}\cdots\sum_{w_N=0}^{\infty}\prod_{i=0}^{N}\left(w_i + \frac{1}{2}\right)\cdot\langle f, P_{w_0}P_{w_1}\cdots P_{w_N}\rangle P_{w_0}P_{w_1}\cdots P_{w_N}. \tag{A.5}$$

Note that $f(z_0, z_1, \ldots, z_N)$ is the probability density function, thus

$$\begin{aligned}\langle f, P_{w_0}P_{w_1}\cdots P_{w_N}\rangle &= \int_D P_{w_0}(z_0)P_{w_1}(z_1)\cdots P_{w_N}(z_N)f\left(z_0, z_1, \ldots, z_N\right)\mathrm{d}z_0\cdots\mathrm{d}z_N \\ &= E[h_1, \ldots, h_N; P_{w_0}(z_0)P_{w_1}(z_1)\cdots P_{w_N}(z_N)].\end{aligned} \tag{A.6}$$

To use the Legendre polynomials as the bases without normalization and avoid computation of the square roots, the spatial Legendre moments are defined as

$$L^{\boldsymbol{T}}_{w_0 w_1 \cdots w_N} = \prod_{i=0}^{N} \left(w_i + \frac{1}{2} \right) \cdot f, P_{w_0} P_{w_1} \cdots P_{w_N} \,, \tag{A.7}$$

which is equivalent to the definition in Equation (4.8).

Furthermore, from Equations (A.5)–(A.7), one can directly derive the expansion series of the probability density function based on the spatial Legendre moments, which appears in Equation (4.10). A similar derivation works for the truncated Legendre polynomial series, since the corresponding function space forms a finite-dimensional subspace of the above Hilbert space.