University of Genoa
Department of Mathematics
PhD in Mathematics and Applications
Curriculum Mathematics Methods for Data Analysis

A Real-Time Adaptive Sampling Strategy Optimized by Uncertainty for Spatial Data Analysis on Complex Domains

Supervisor: Professor Marino Vetuschi Zuccolini
Department of DISTAV, University of Genoa

Supervisor: Dr. Michela Spagnuolo
Consiglio Nazionale delle Ricerche
Istituto di Matematica Applicata e Tecnologie Informatiche

PhD Student: Serena Berretta
Freshman Number: 3508848

Academic Year 2020/2021

Contents

Introduction

Environmental monitoring is used to reveal the state of the environment, to inform experts and help them to prioritise actions in the context of environmental policies. Environmental monitoring deals with the control of pollutants in air, water and soil, the identification of risks related to climate change and anthropic impacts, the early detection of natural disasters and hazards. Basically, this means observing and characterising a phenomenon, which happens in a time and space range, to interpret it and foresee how it will evolve. For this, it is strategic to provide a reliable representation of a phenomenon in its geographical context.

Environmental sampling is the way the environment is interrogated to get measures of environmental (e.g., physical, chemical) parameters in a limited set of locations (samples). The environmental properties varies from place to place in continuum and there are infinitely many places at which we might record what they are like, but practically we can measure them at only a finite number by sampling. The role of the location in which samples are collected is very crucial.

Typically, sample locations are pre-defined in the survey design phase, before the start of the sampling, and placed either randomly or on a regular grid or along directions that are selected with respect to any a priori knowledge of the expert. Some traditional environmental surveys are based on the collection of physical samples of water, air or soil that are finally analysed in laboratory after the conclusion of the sampling campaign. No feedback is available during the survey, with no possibility to adapt the investigation according to any insight that might emerge. Clearly, this procedure could be very expensive in time and cost and results could be very unreliable.

The focus of the thesis is the study of a mathematical framework that supports a reasoned and non-random sampling of environmental variables, with the aim of defining a methodological approach to optimise the number of sampling required while maintaining a target precision. The arrangement of points is not selected or known a priori; conversely, we propose an iterative process where the next-sample location is determined on-the-fly on the basis of the environmental scenario that is

delineated more and more accurately at each iteration.

This methodological approach may have an important application impact: leveraging on the use of autonomous platforms, equipped with sensors able to capture and analyse in real-time physical or chemical parameters, it is possible to build *in-situ* dynamic and real-time surveying campaign that does not need a laboratory analysis process, reducing highly the time and costs of the survey. This aspect has been also experimented during the thesis research programme, with interesting results demonstrating the innovation potential of the method proposed.

It remains to individuate the driving dimension the system should adapt to control the iterative process, which is one of the distinctive feature of approach studied. The idea is to rely on the uncertainty that the evolving map of the reconstructed parameters exhibits. At each iteration, the distribution map is updated with the new incoming data. The geostatistical analysis we implement provides a predicted value and the related uncertainty about that value, actually providing an uncertainty map beside the predicted distribution. The system responds to the current state by requiring a measurement in the area with highest uncertainty, to reduce uncertainty and increase accuracy.

At the theoretical level, the whole approach has been studied introducing innovation at two stages: first, introduction of unstructured grids to discretise the physical domain subject to the sampling process, and second, study of a suitable change of support model to sustain a correct estimation of the reconstructed parameter map that works on unstructured grids.

Unstructured grids are nowadays well studied and methods exist to use them efficiently. Environmental survey areas to monitor are often characterised by very complex boundaries and, most importantly, adaptive resolutions are useful to adapt the resolution of the discretisation to represent some important areas with a finer resolution leaving coarser resolution in less important areas. Unstructured grids are more flexible to faithfully represent complex geometries compared to structured grids. The latter would need of very high-resolution model made of millions or even billions of small elements to fit the geometries of the survey area, requiring high-performance computers to be handled efficiently. The blocks of structured grids have indeed all the same volume, so preventing an adaptive resolution of the domain. Unstructured grids have been considered in this thesis, in particular those composed by tetrahedra of different sizes, since, at the state of the art, they are the best option to model a complex domains.

The usage of unstructured grids introduces the second innovation aspect studied in the thesis, which is the change of support model. Measurements of environmental

variables are modelled mathematically as samples represented on a point support, and from this model often one is interested to make estimation on large supports (blocks). In environmental survey, the distribution of the variables on these blocks is the quantity of interest and must be determined from the core samples. While in the literature there is a lot of work done in regular block structures, we have studied the consequences of irregular block structures and dimensions and we have defined a change of support model suitable to unstructured grids.

An important contribution of this thesis is also the development of several geostatistical functionalities to be included in a spatial data analysis software and the implementation of a graphical user interface that allows the user to monitor and interpret the sampling and reconstruction process of the environmental status in real time. All the results of this thesis are obtained and visualised using these tools.

The thesis consists of five chapters. Chapter 1 reviews the essential theory on spatial data analysis which is instrumental to the following chapters. It presents the fundamental concepts of geostatistics to provide an estimation of a spatial phenomenon. In this chapter both theoretical description and implementation of geostatical methods are provided. For this, at the end of this chapter algorithms to implement in *C++* the theoretical aspects are presented.

In Chapter 2 change of support models are introduced with a particular focus on Discrete Gaussian Model (DGM). Details on the theoretical formulation and definition of the DGM are presented. Then, it is described how the DGM is extended to deal with unstructured grids. At the end of this chapter the implementation of the theoretical aspects is shown.

During the implementation procedure of the DGM, a crucial step is the computation of the block-to-block covariance [Chilès and Delfiner, 2012]. Chapter 3 presents a new method to compute it when blocks are tetrahedra. A comparison between several methods to generate points in a volume is presented and the extension to the tetrahedral support is one of the main development in this thesis. At the end of this chapter a testing procedure to evaluate results is conducted.

Chapter 4 describes the new adaptive sampling strategy to collect data in order to give a representation of a spatial phenomenon. The chapter presents the optimisation of this iterative process using an uncertainty measure related to the estimates of the environmental variable. At the end of the chapter, a comparison between our proposal strategy of adaptive sampling and other sampling strategies is presented and the results show interesting improvements using the new approach.

Finally, Chapter 5 discusses an application to a real case of an environmental monitoring survey in the framework of the project MATRAC-ACP whose aim was

to increase the protection of marine waters in ports using new technologies and new adaptive sampling methodologies. The work done within the scope of this project is particularly relevant as it allowed us to validate the whole theoretical framework in a real context, showing the efficacy of the adaptive sampling strategy proposed as the core engine of a novel surveying approach.

The main contributions of this thesis are:

- the definition of a new adaptive sampling strategy optimised by uncertainty to provide an estimation of the spatial distribution of an environmental variable (and of its uncertainty) which balances reliability and efficiency;
- the extension of the estimation procedure in order to deal with unstructured grids when the domain to be considered is complex or an adaptive resolution is required;
- a new method to generate points inside tetrahedra and also compute the block-to-block covariance when a tetrahedral support is considered in the DGM implementation;
- the implementation in *C++* of new algorithms and graphical user interfaces for spatial data analysis; in particular a new function to fit the variogram in an automatic way and new functions to implement change of support model in the spatial analysis.

Chapter 1

Spatial Data Analysis

This chapter introduces the main concepts of geostatistics that will be used throughout the thesis for defining the spatial data analysis framework which is the focus of the research undertaken.

In Section 1.1 the framework of spatial data analysis is explained and some definitions needed to lay the foundation for the formalism of this thesis are shown. In Section 1.2 one of the main tool for geostatistics is described: the variogram; here the theoretical aspects of its estimation and modeling are explained. In Section 1.3 the kriging procedure for the interpolation and estimation of environmental variables on the basis of sampled data is described. Finally, in Section 1.4 the simulation approach is explained, in particular the Sequential Gaussian Simulations, which is the basis of the adaptive sampling strategy defined in the thesis.

The chapter ends with Section 1.5, where algorithmic aspects addressed during the thesis are introduced. The theoretical tools most relevant to the research work have been implemented in *C++* code with the aim of building a geostatistical toolkit supporting the experimentation of the spatial data analysis framework devised. In particular, a new procedure of the automatic fitting of the experimental variogram is described.

1.1 Geostatistics

Environmental properties have arisen as the result of actions and interactions of many different processes and factors. The environment is the outcome of these processes and varies place to place with a great complexity and at many spatial scales, from micrometres to hundreds of kilometres. An additional feature of the environment is that at some scale the values of its properties are positively related (autocorrelated). Places close to one another tend to have similar behaviour,

whereas ones that are farther apart differ more on average. Geostatistics expresses this intuitive knowledge quantitatively and then uses it for prediction. There is inevitably error in the estimates, but by quantifying the spatial autocorrelation at the scale of interest errors can be minimised [Webster and Oliver, 2007].

Geostatistics aims at providing quantitative descriptions of spatial variables distributed in space using information about their autocorrelation. Since the environment and its attributes result from many physical and biological processes that interact in a chaotic way, the outcome is so complex that the variability appears to be random. This complexity and the incomplete understanding of the processes mean that mathematical functions are not adequate to describe this environmental properties. Consequently, a fully deterministic solution seems not the best way. The aim is to describe quantitatively how soil varies spatially and to predict its values at unsampled locations. In addition, estimates of the errors on these predictions is useful to evaluate their reliability. Therefore, the solution seems to lie in a probabilistic or stochastic approach [Webster and Oliver, 2007].

Some sort of spatial relationship between sampled values is assumed in order to produce an estimate for a value at a specified location. Basically, all mapping methods assume that if locations are close together then values will be close together. Conceptually, an estimator put together from neighbouring samples will be more useful than one which includes more distant samples.

1.1.1 Random Field

Given a domain $D \subset \mathbb{R}^d$ (with a positive volume) and a probability space (Ω, A, P), a random function is a function of two variables $Z(x, \omega)$ such that for each $x \in D$ the section $Z(x, \cdot)$ is a random variable on (Ω, A, P) [DeGroot and Schervish, 2012]. Each of the functions $Z(\cdot, \omega)$ defined on D as the section of the random function at $\omega \in \Omega$ is a realization of the random function. For short the random function is simply denoted by $Z(x)$ [Chiles and Delfiner, 2009]. In the literature a random function is also called a stochastic process when x varies in a 1D space, and it is called a random field (RF) when x varies in a space of more than one dimension $(d > 1)$.

In geostatistics, the study of the behaviour of a variable $z(x)$, which is called regionalized variable, is considered as the realization of a random function $Z(x)$.

A random function is described by its finite-dimensional distributions, namely the set of all multidimensional distributions of k-tuples $(Z(x_1), Z(x_2), \ldots, Z(x_k))$ for all finite values of k and all configurations of the points $x_1, x_2, \ldots, x_k$. A complete description of a random field consists in defining all of its finite-dimensional distri-

butions, which can be done by means of the multivariate cumulative distribution functions (cdf) [DeGroot and Schervish, 2012]. Thus, in order to describe the RF $Z(x)$, for every k and every set of points $x_1, \dots x_k \in D$ the finite-dimensional cdf should be given by

$$F_{x_1,\dots,x_k}(z_1, \dots, z_k) = P[Z(x_1) \leq z_1, \dots, Z(x_k) \leq z_k].$$

This is called spatial distribution [Chiles and Delfiner, 2009].

In practice, defining a complete set of finite-dimensional cdfs is almost never possible, except for the few known analytical models such as multivariate Gaussian RF. For that reason, the random fields in geostatistics are often characterised only with respect to the first and second order moments [DeGroot and Schervish, 2012]. Thus, defining a RF reduces to defining the marginal distribution $F_x(z) = P[Z(x) \leq z]$ at every point $x \in D$ and defining the covariance function $C(x, x^{'}) = Cov[Z(x), Z(x^{'})]$ for all pairs of points $x, x^{'} \in D$. Certainly, it is necessary to assume the existence of the first and second order moments for $Z(x)$. This definition through the marginal distribution and covariance in the general case does not determine a random field in a unique manner. [Chiles and Lantuéjoul, 2005] demonstrate three different random set models with the same bivariate and even trivariate distributions. They conclude that, when a RF taken into account only considering the marginal distribution and the covariance function, classes of equivalence of RF are considered.

A particular case of great practical importance is when the finite-dimensional distributions are invariant under an arbitrary translation of the points by a vector h (strict stationarity)

$$P[Z(x_1) < z_1, \dots, Z(x_k) < z_k] = P[Z(x_1 + h) < z_1, \dots, Z(x_k + h) < z_k]$$

Such RF is called stationary. Physically, this means that the phenomenon is homogeneous in space.

When the random function is stationary, its moments, if they exist, are obviously invariant under translations. Considering points x and $x + h$ of $\mathbb{R}^d$, the first two moments are:

$$E[Z(x)] = E[Z(x + h)] = m, \tag{1.1}$$

$$E[(Z(x) - m)(Z(x + h) - m)] = C(h) \tag{1.2}$$

The mean is constant and the covariance function only depends on the separation h. Then, by definition, a random function satisfying the above conditions is second order stationary (or weakly stationary). The abbreviation SRF will designate a

second-order stationary random field. An SRF is isotropic if its covariance function only depends on the length $|h|$ of the vector h and not on its orientation.

A milder hypothesis is to assume that for every vector h the increment $Z(x+h) - Z(x)$ is an SRF in x. Then $Z(x)$ is called an intrinsic random function (IRF) and is characterised by the following relationships:

$$E[Z(x+h) - Z(x)] = a_h, \tag{1.3}$$

$$Var[Z(x+h) - Z(x)] = 2\gamma(h) \tag{1.4}$$

a_h is the linear drift of the IRF (drift of the increment) and $\gamma(h)$ is its variogram function (Section 1.2.1). If the linear drift is zero ($a_h = 0$) - that is, if the mean is constant - this is the usual form of the intrinsic model.

Gaussian Random Field

A RF is Gaussian if all its finite-dimensional distributions are multivariate Gaussian. Since a Gaussian distribution is completely defined by its first two moments, knowledge of the mean and the covariance function suffices to determine the spatial distribution of a Gaussian RF. In particular, second-order stationarity is equivalent to full stationarity. A Gaussian IRF is an IRF whose increments are multivariate Gaussian. A weaker form is when only the marginal distribution of $Z(x)$ is Gaussian. This by no way implies that $Z(x)$ is a Gaussian RF, but this assumption is sometimes made.

1.2 Spatial Distribution

In Section 1.1 it was shown that a stationary random function (SRF) $Z(x)$ is characterised by its mean (Formula 1.1) and its covariance function (Formula 1.2). A related function is the correlogram $\rho(h) = \frac{C(h)}{C(0)}$, which is the correlation coefficient between $Z(x)$ and $Z(x+h)$. The covariance and the correlogram show how this correlation evolves with the separation, or lag, h. Note that h is a vector. These functions therefore depend both on its length, which is the distance between x and $x+h$, and on its direction. When the covariance depends only on distance, it is said to be isotropic. A covariance is an even function: $C(h) = C(-h)$; and it is bounded by its value at the origin (i.e., the variance of the SRF): $|C(h)| < C(0)$. [Webster and Oliver, 2007]

1.2.1 Variogram Estimation

An intrinsic random function (IRF) is a random function whose increments are second-order stationary. It is characterised by its linear drift (Formula 1.3) and its variogram (Formula 1.4). The variogram shows how the dissimilarity between $Z(x)$ and $Z(x+h)$ evolves with the separation h. Like the covariance, it can be isotropic or anisotropic. The variogram function's properties are:

- even, $\gamma(h) = \gamma(-h)$,
- no negative, $\gamma(h) > 0$
- $\gamma(0) = 0$.

An SRF is obviously also an IRF and therefore has a variogram. In that case the variogram is linked to the covariance by the relation

$$\gamma(h) = C(0) - C(h) \tag{1.5}$$

Thus, the variogram of an SRF is bounded by $2*C(0)$. Formula 1.5 shows that if the covariance is known, the variogram is also known. The relation between variogram and covariance is shown in Figure 1.1(a).

Unless a SRF with a known mean is considered, an exceptional situation, there are two reasons to favor the variogram over the covariance. The first is theoretical: since the class of IRFs includes the SRFs, the variogram is a more general tool than the covariance. The second reason is practical: the variogram does not require the knowledge of the mean, whereas to compute the covariance the mean has to be estimated from the data, which introduces a bias. This bias cannot be corrected unless the covariance function, or at least the correlation function, is already known, which is not the case (unless the data can be considered uncorrelated, but this is a very special case). For these reasons, it is almost always used the variogram in the spatial analysis.

Let us consider a regionalized variable $\{Z(x), x \in D \subset \mathbb{R}^d\}$, with a set of known values at N sample locations $\{z(x_\alpha); x_\alpha \in D, \alpha = 1, \ldots, N\}$. Physical intuition suggests that two points that are close assume close values because these values were generated under similar physical conditions. On the other hand, at long distances the genetic conditions are different, and greater variations are to be expected. This intuition of variability with distance can be quantified using the variogram.

The variogram can be defined directly: denoting by N_h the count of pairs of

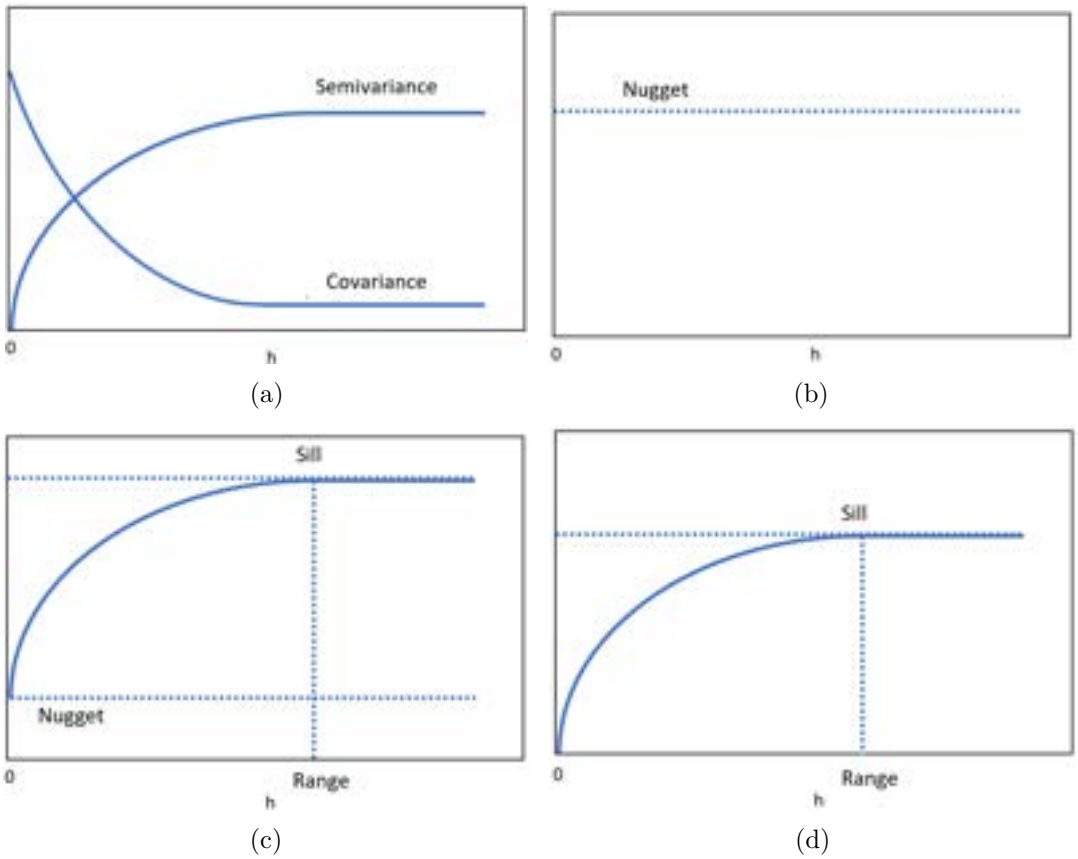


Figure 1.1: (a): the graphical relation between Semivariance and Covariance is shown; (b): the variogram of a pure nugget distribution is shown; (c): a variogram with all three parameters (Sill, Range and Nugget) is shown; (d): a variogram without nugget variance is shown.

points separated (approximately) by the lag h, variogram is defined by

$$\gamma(h) = \frac{1}{2N_h} \sum_{x_\beta - x_\alpha \approx h} (Z(x_\beta) - Z(x_\alpha))^2 \tag{1.6}$$

The graph of the sample variogram $\gamma(h)$ against $|h|$, generally shows the following behaviour:

- it starts at zero (for $h = 0$, $Z(x + h) - Z(x) = 0$);
- it increases with $|h|$;
- it continues to increase, or else stabilizes at a certain level.

The rate of the variogram increase reflects the degree of dissimilarity of ever more distant samples. The variogram can increase indefinitely if the variability of the phenomenon has no limit at large distances. If, conversely, the variogram stabilizes at a value, called the *sill*, it means that there is a distance beyond which $Z(x)$ and $Z(x + h)$ are uncorrelated. This distance is called the *range* (Figure 1.1(d)). Moreover, it is interesting to examine the variogram behaviour near the origin because it reflects the continuity and the spatial regularity of the regionalized variable. Very often, $\gamma(h)$ does not seem to tend to zero when $h \to 0$ (discontinuity at 0); this is called *nugget effect*, (Figure 1.1(c)). This means that the regionalized variable is generally not continuous. The origin of this denomination is as follows: in gold deposits, gold commonly occurs as nuggets of pure metal that are much smaller than the size of a sample. This results in strong grade variability in the samples, even when physically very close and therefore in a discontinuity of the variogram at the origin [Matheron, 1963]. By extension, the term "nugget effect" is applied to all discontinuities at the origin. In general, the nugget effect is due to:

- a phenomenon with a range shorter than the sampling support (true nugget effect, Figure 1.1(b))
- measurement or positioning errors.

In environmental applications the presence of nearby points characterised by a very different behaviour is not so rare. The estimator of the variogram could be sensitive to the presence of outliers data, causing a wrong estimation of the variogram. This will result in incorrect variances when the value of an environmental property is estimated. In the literature, several robust estimators of the variogram have been proposed as improvements (e.g. [Cressie and Hawkins, 1980], [Genton, 1998] and [Omre, 1984]). A comparison of some robust estimators of the variogram in soil

survey is included in [Lark, 2000]. They seem to be suitable for analysis of soil data in circumstances where the standard estimator is likely to be affected by outliers. However, robust statistical methods will not yet be integrated into the methodology proposed in this thesis, but they will be considered for future improvements.

1.2.2 Variogram Modeling

When an empirical variogram is computed, an ordered set of values consisting of $\{\gamma(h_1), \gamma(h_2), \dots\}$ at particular lags $h_1, h_2, \dots$ is obtained. The ensemble of the pairs of points $\{(\gamma(h_j), j = 1, 2, \dots\}$ summarizes the spatial relations in the data. The true variogram representing the regional variation is continuous, and it is this variogram that one is really interested to know. The observed values are used as approximations to the function by imagining a curve passing through them. Therefore, the aim is to obtain a way to describe the spatial relations of the whole region of interest: the fitting of a continuous function is able to describe the spatial variation so it is possible to estimate or predict values also at unsampled place. For this, semivariances at lags for which we have no direct comparisons is required, calculating these from such a function. The function must therefore be mathematically defined for all real h. There are a few principal features that a function must be able to represent. These include:

- a monotonic increase with increasing lag distance;
- a constant maximum or asymptote (*sill*);
- a positive intercept on the ordinate (*nugget*);
- periodic fluctuation, or a 'hole';
- anisotropy.

There are several functions that encompass the features listed above. The main functions considered in this thesis are listed below.

Spherical Model

The spherical function is one of the most frequently used models in geostatistics, in one, two and three dimensions. The variogram is defined as

$$\gamma(h) = \begin{cases} c\left(\frac{3h}{2a} - \frac{1}{2}\left(\frac{h}{a}\right)^3\right) \\ c \end{cases} \tag{1.7}$$

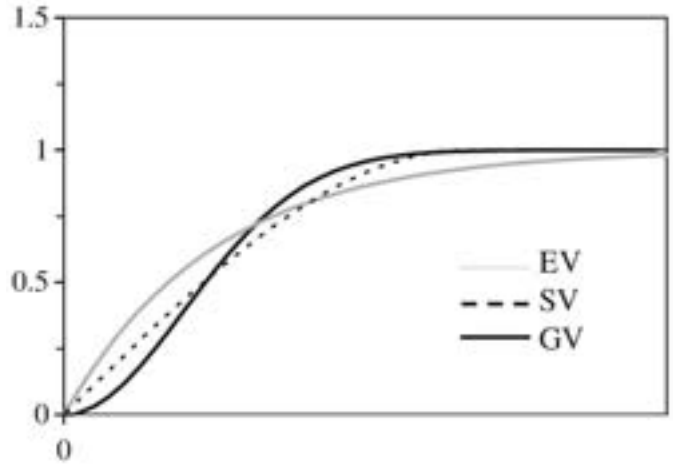


Figure 1.2: Comparison among variograms of Sperical model (SV), Exponential model(EV), Gaussian model (GV).

where c is the sill variance and a is the range.

Exponential Model

A function that is also much used in geostatistics is the negative exponential:

$$\gamma(h) = c\left(1 - e^{-\frac{h}{r}}\right) \tag{1.8}$$

with sill c, and a distance parameter, r, that defines the spatial extent of the model. The function approaches its sill asymptotically, and so it does not have a finite range. Nevertheless, for practical purposes it is convenient to assign it an effective range, and this is usually taken as the distance at which γ equals 95% of the sill variance, approximately $3r$. Its slope at the origin is $\frac{c}{r}$.

Gaussian Model

Another function with reverse curvature near the origin recurs again and again in geostatistics is the so-called Gaussian model with equation:

$$\gamma(h) = c\left(1 - e^{-\frac{h^2}{r^2}}\right) \tag{1.9}$$

A graphical representation of these three variogram models is in Figure 1.2.

The models described above are those that are commonly used for variograms in environmental surveys. Actually the task is to fit them to the experimental values. In general, there are still practitioners who fit the models by eye, but for an automatic procedure, not conditioned by an user, it is convenient fit models numerically and

automatically. In this thesis an algorithm to interpolate the experimental variogram automatically is proposed and described in Section 1.5.1.

1.2.3 Directional Variogram

When the variogram does not vary with direction, it is said to be isotropic. It is then a function of the modulus of the lag, $|h|$, namely of the distance between the points. On the other hand, the variogram can show different behaviours along the different directions of the separation $\vec{h}$, namely display an anisotropy. A phenomenon is said to be anisotropic when its pattern of spatial variability changes with direction. This is frequent in 2D, and especially in 3D where vertical variability is rarely of the same nature as horizontal variability. For simplicity of presentation, in the following, reference will be made to the two-dimensional anisotropy models with vector $\vec{h} = (h_x, h_y)^T$. [Isaaks and Srivastava, 1989] discusses the three-dimensional anisotropic models.

Two typical cases of anisotropy can occur: the geometric anisotropy and the zonal anisotropy.

Geometric Anisotropy

An anisotropy is said to be geometric when: *(i)* the directional semivariograms have the same shape and sill but different range values and *(ii)* the plot of range values versus the azimuth θ of the direction is an ellipse. In this plot the major axis of the ellipse, corresponding to the direction of maximum continuity, forms an angle θ with the north direction (y-axis). The azimuth angle θ is measured in degrees clockwise from the y-axis. The minor direction of anisotropy is perpendicular to the major axis of the ellipse and has an azimuth $\theta + 90°$

The major and minor ranges of anisotropy a_θ and $a_{\theta+90°}$ are plotted as the major and minor radii of the ellipse. The anisotropy factor λ is defined as the ratio of the minor range to the major range, $\lambda = \frac{a_{\theta+90°}}{a_\theta} < 1$.

The anisotropy correction consist of trasforming the vector of original coordinates $\vec{h} = (h_x, h_y)^T$ into a new vector $\vec{h}' = (h'_{\theta+90°}, h'_\theta)^T$, so that the value of the anisotropic variogram model $\gamma(\vec{h})$ identifies that of an isotropic model $\gamma^t(|h'|)$ in the new system of coordinates:

$$\gamma(\vec{h}) = \gamma^t(|h'|) \qquad \text{with } |h'| = \sqrt{{h'_{\theta+90°}}^2 + {h'_\theta}^2}$$

where $\gamma^t(\cdot)$ is an isotropic model with a range equal to the minor range of anisotropy, $a_{\theta+90°}$.

The coordinate transformation calls for two key parameters: the azimuth angle θ of the direction of maximum continuity and the anisotropy factor λ. The transformation proceeds in two steps:

1. the coordinate axes are rotated clockwise so as to identify the main axes of the ellipse. The rotation angle corresponds to the azimuth θ. The new vector of coordinates is:

$$\vec{h}_\theta = \begin{bmatrix} h_{\theta+90^\circ} \\ h_\theta \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \cdot \begin{bmatrix} h_x \\ h_y \end{bmatrix}$$

2. The ellipse is then rescaled to a circle of radius equal to the minor range $a_{\theta+90^\circ}$. The rescaling of the new coordinates $(h_{\theta+90^\circ}, h_\theta)^T$ is written as:

$$\vec{h}' = \begin{bmatrix} h'_{\theta+90^\circ} \\ h'_\theta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \lambda \end{bmatrix} \cdot \begin{bmatrix} h_{\theta+90^\circ} \\ h_\theta \end{bmatrix}$$

Any isotropic model can be considered as a particular case of the geometric anisotropic model where the anisotropy factor is equal to one ($a_{\theta+90^\circ} = a_\theta$)

Zonal Anisotropy

An anisotropy that involves sill values varying with direction is said to be zonal. The semivariogram in the direction of azimuth ϕ has a longer range a_ϕ and also a larger sill than in other directions. Such anisotropy can be modeled as the sum of an isotropic transition model $\gamma_1(|h|)$ and a zonal model $\gamma_2(h_\phi)$ which depends only on the distance in the direction of greater variance

$$\gamma(\vec{h}) = \gamma_1(|h|) + \gamma_2(h_\phi)$$

where the model $\gamma_2(\cdot)$ has a range a_ϕ.

The component $\gamma_2(h_\phi)$ can be seen as an extreme case of the geometric anisotropic model and its modeling proceeds in two steps:

1. Rotate clockwise the coordinate axes so that the y-axis identifies the direction of maximum continuity defined as the direction perpendicular to that of greater variance (highest sill). The new vector of coordinates (h_ϕ, h_θ) is computed as

$$\vec{h}_\theta = \begin{bmatrix} h_\phi \\ h_\theta \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \cdot \begin{bmatrix} h_x \\ h_y \end{bmatrix}$$

where θ is equal to $\phi - 90°$.

2. The new axes are then rescaled so that the zonal model does not contribute to the direction of maximum continuity (azimuth θ). Such rescaling amounts to setting the range a_θ in that direction to infinity, hence the anisotropy factor λ to zero:

$$\vec{h}' = \begin{bmatrix} h'_\phi \\ h'_\theta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} h_\phi \\ h_\theta \end{bmatrix}$$

The zonal anisotropy is not nearly as common as the geometric anisotropy.

In practice, the experimental variogram is calculated by classes of directions. The main anisotropy directions are often suspected from geological knowledge, and the variogram is computed along these directions. If this is not the case, it is necessary to compute the variogram in several directions to detect a possible anisotropy.

In order to evaluate the anisotropy and to compute the directional variograms, some directional information is needed. These parameters are a set of directions, β, and a range in direction, α, such that $\alpha = \frac{\pi}{n_{dir}}$, where n_{dir} is the number of directions and β progresses in steps of α from 0 to $\frac{\pi(n_{dir}-1)}{(n_{dir})}$. For example, if we choose four directions ($n_{dir} = 4$) then a progression for β would be $0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$ i.e. $0, 45°, 90°, 135°$ degrees, with a $\alpha = \frac{\pi}{4}$ (45°). This ensures complete coverage and no overlap between the different directions. For a point x_i and a second $x_i + \vec{h}$ within the zone defined by β and α, the value $\left[Z(x_i) - Z(x_i + \vec{h})\right]^2$ contributes to $\hat{\gamma}(\vec{h}) = \hat{\gamma}(h, \beta)$. When all comparisons have been made the experimental variogram will consist of the set of averages for the nominal lags in both distance and direction. The omnidirectional variogram is a special case of the directional variogram when $\alpha = 180$.

Our algorithm for the directional variograms is in progress and future works involve the integration of the anisotropy in the estimation analysis. At the moment we are able to compute the experimental variograms at several directions (n_{dir}). Two examples of directional variograms computed on simulated data are shown in Figure 1.3 and 1.4 varying the number of directions. In the first example with $n_{dir} = 2$ the spherical model is fitted on the two experimental variograms with approximately the same range. In this case the RF can be assumed to be isotropic. On the other hand, in the second example, with $n_{dir} = 4$, there seems to be an anisotropic behaviour. The range in the directions 45° and 90° is bigger then that in the directions 0° and 135°. An issue of the directional variograms is highlighted in this second example:

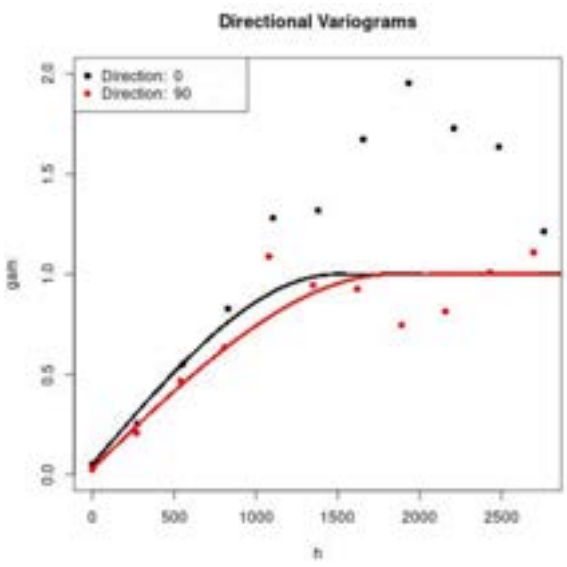


Figure 1.3: Directional variograms with $n_{dir} = 2$, $\beta = (0°, 90°)$ degrees and $\alpha = 90°$.

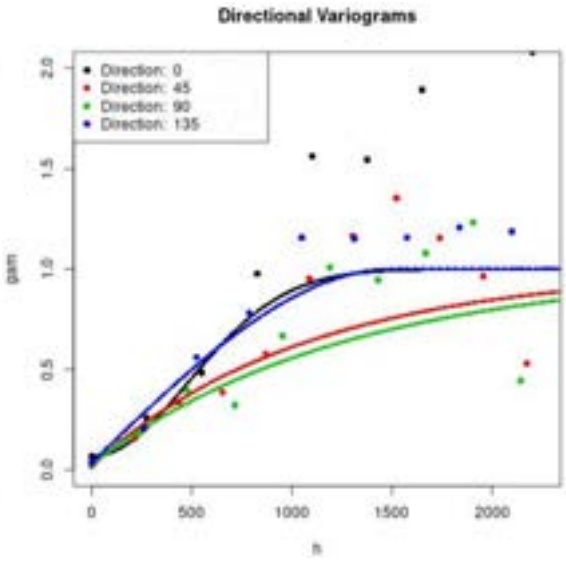


Figure 1.4: Directional variograms with $n_{dir} = 4$, $\beta = (0°, 45°, 90°, 135°)$ degrees and $\alpha = 45°$.

the fitted model on the experimental variogram computed on the direction of 0° is Gaussian instead of spherical as happens for the other directions. There should be consistency among the type of fitted model on the several directional variograms. A solution to deal with this issue will be studied.

Moreover, it is important to note that when the direction is considered in spatial data analysis the number of data to estimate $\hat{\gamma}(\vec{h})$ may be very low and in these cases one must pay attention.

When the spatial distribution is anisotropic and there is a direction of maximum continuity, where the range is bigger then the range in the perpendicular direction (geometric anisotropy), then two pairs of points at the same distance in module, $|h|$, could have a very different value of $\left[Z(x_i) - Z(x_i + \vec{h})\right]^2$ if they are aligned on different directions, $\vec{h}$. We will see in Chapter 2 that in some applications, this can be crucial.

1.2.4 Variogram with Geodetic Distance

The distance between two points is the length of the path connecting them [Berg et al., 1997]. In general, the distance between points $\mathbf{s}$ and $\mathbf{t}$ in an Euclidean space $\mathbb{R}^d$ is given by

$$D = |\mathbf{t} - \mathbf{s}| = \sqrt{\sum_{i=1}^{d} |t_i - s_i|^2}$$

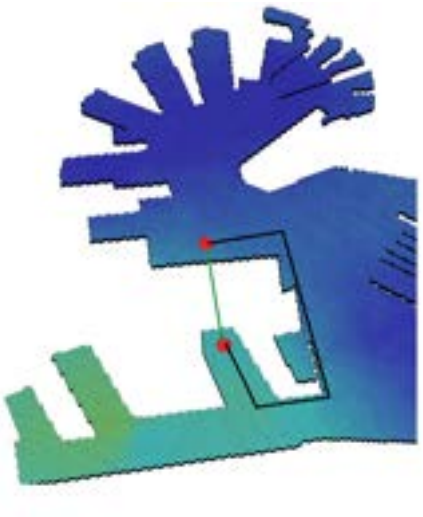


Figure 1.5: The red dots are the samples collected from the two sides of the pier. The green line corresponds to the Euclidean distance while the black one corresponds to the geodetic distance.

In particular, in the Euclidean three-dimensional space, the distance between points $s = (s_x, s_y, s_z)$ and $t = (t_x, t_y, t_z)$ is

$$D = \sqrt{(t_x - s_x)^2 + (t_y - s_y)^2 + (t_z - s_z)^2} \tag{1.10}$$

The Euclidean distance is generally used for the computation of the experimental variogram. However, the domain subject to surveying can be geometrically very complex and can include several obstacles, which define a geometrically non convex-domain. In this cases, the Euclidean distance between two points is not representative of the distance between the two points in the real physical domain: the physical distance is indeed an important factor to capture the correlation between two measures, and this distance should reflect the distance within the domain and not the distance of "short-cuts" measuring outside the domain.

Two points that are separated by an obstacle could be very close considering the Euclidean distance as in Equation 1.10, but in reality, to connect the two points physically, the path can be much longer. For example, in the framework of marine monitoring in ports, two points that are on opposite sides of a pier (Figure 1.5) would be very close using the Euclidean distance that does not consider the pier and passes through it. Their sampled values could also be very different if on one side of the pier there was a drain. In the calculation of the variogram these values could lead to an estimation error with an overestimation of the nugget and / or underestimation of the range. In reality, the two points are much more distant and the so different values of the variable of interest would not be misinterpreted if

Figure 1.6: The shortest (geodesic) path that connects five samples in the 3D model of the Genoa harbor.

another metric were used to measure their distance. This can be solved using the geodesic distance.

The Discrete Geodesic Problem [Mitchell et al., 1987] is the problem of finding shortest paths between pairs of points on the surface of a 3D polyhedron such that the path, termed geodesic path, is constrained to lie on the surface of the polyhedron. Let τ be a polyhedral surface in $\mathbb{R}^3$ and let s and t be two points on the surface of τ. The discrete Geodesic Problem is to find the shortest path in the Euclidean Metric from s to t such that the shortest path is constrained to lie on the surface of τ.

Definition 1. (**Geodesic distance**) The Geodesic distance is defined as

$$\forall (s,t) \in \tau \subset \mathbb{R}^3, \qquad d_\tau(s,t) = \min_{\delta \in \Delta(s,t)} L(\delta) \tag{1.11}$$

where $\Delta(s,t)$ denotes the set of piecewise smooth curves joining s and t and $L(\delta)$ is the length of the curve δ [Peyré and Cohen, 2009].

In Figure 1.6 is shown an example in the 3D model of Genoa harbor of the shortest path (geodesic) used to connect five samples.

Numerical computation of shortest paths or geodesics on curved domains, as well as the associated geodesic distance, arises in a broad range of scientific applications (e.g. digital geometry processing, computer graphics). Relative to Euclidean distance computation, these tasks are complicated by the influence of curvature on the behaviour of shortest paths. In spite of the difficulty of this problem, recent literature has developed a wide variety of sophisticated methods that enable rapid queries of geodesic information, even on relatively large models. A survey reviews of the major categories of approaches to the computation of geodesic paths and distances is presented in [Crane et al., 2020].

The geodesic distance is already used in the spatial analysis: [Grazzini et al., 2007]

and [Banerjee, 2005] use the geodesic distance for the spatial interpolation and spatial modeling.

Adopting geodesic distance as an element of a real-time sampling strategy permits to devise general strategy suitable to any geometry of the physical domain to sample but, at the same time, introduces the need to plan carefully the algorithmic side of the geoedesic distance computation, to avoid bottle necks in terms of computational time.

1.3 Kriging Interpolation

Geostatistics has the task of providing values of the environmental properties even where the samples are not recorded, it works through an estimation of them with interpolation methods. The aim is the estimation of a variable of interest over a domain on the basis of values observed at a limited number of points. From a deterministic viewpoint this is an interpolation problem. The variable of interest is approximated by a parametric function where the parameters are selected so as to optimise some criterion of best fit at the data points. Once the approximating function is determined, it is a simple matter to evaluate it wherever needed.

Another possibility is a probabilistic approach known as kriging. It produces an interpolation function based on a covariance or variogram model derived from the data. The basic idea of kriging is to predict the value of a function at a given point by computing a weighted average of the known values of the function in the neighborhood of the point.

Let us consider a regionalized variable $\{Z(x), x \in D \subset \mathbb{R}^d\}$, with a set of known values $\{z(x_\alpha)\}$ at N sample locations $S_N = \{x_\alpha, \alpha = 1, \ldots, N\}$.

The kriging estimator of $Z(x_0)$ is of the form:

$$\hat{Z}(x_0) = \sum_{\alpha=1}^{N} \lambda_\alpha Z(x_\alpha) + \lambda_0$$

where λ_α is a weight placed on $Z(x_\alpha)$ and λ_0 is a constant that depends on x_0.

The summation is extended over all α indexes in S_N. In practice, N may be too large to allow computation and a "moving neighborhood" or "local neighborhood" has to be used, including only a subset of the data for the estimation of each target location.

1.3.1 Ordinary Kriging

In most practical situations the mean $m(x)$ is not known. The simplest case is when the mean is a constant $m(x) = a_0$ and leads to ordinary kriging (OK). It was developed by Matheron in [Matheron, 1963] and is the form of kriging used most because it works under simple stationarity assumptions and does not require knowledge of the mean.

The aim is to estimate $Z_0 = Z(x_0)$ from N observations $\{z(x_\alpha), \alpha = 1, \ldots, N\}$, using the estimator $\hat{Z} = \sum_\alpha \lambda_\alpha Z_\alpha + \lambda_0$ where the constant λ_0 and the weights λ_α are selected so as to minimise in the model the expected mean square error (m.s.e.):

$$E[\hat{Z} - Z_0]^2 = Var[\hat{Z} - Z_0] + \left[\lambda_0 + \left(\sum_\alpha \lambda_\alpha - 1\right) a_0\right]^2$$

Only the bias term on the right-hand side involves λ_0, but we cannot minimise it without knowledge of a_0. An intuitive solution would be to replace a_0 by an estimate $\hat{a}_0$ and solve for λ_0, but this estimate would necessarily depend on the data so that λ_0 would no longer be a constant. The only real solution is to set $\lambda_0 = 0$ and impose the condition $\sum \lambda_\alpha - 1 = 0$ on the weights λ_α. The bias $E[\hat{Z} - Z_0]$ is then zero whatever the unknown constant a_0. The consequence for not knowing the mean is to restrict ourselves to a linear estimator with weights adding up to 1. Subject to this condition the m.s.e. is equal to the variance of the error $\hat{Z} - Z_0$ and depends only on covariances:

$$Var[\hat{Z} - Z_0] = \sum_\alpha \sum_\beta \lambda_\alpha \lambda_\beta \sigma_{\alpha\beta} - 2 \sum_\alpha \lambda_\alpha \sigma_{\alpha 0} + \sigma_{00}$$

where $[\sigma_{\alpha\beta}]$ is the $N \times N$ matrix of data-to-data covariances and $[\sigma_{\alpha 0}]$ is the $N-$vector of covariances between the data and the target point x_0 and σ_{00} is the variance of Z_0. These variances can be calculated with the variogram using the relation described in Section 1.2.1.

The problem can now be reformulated as follows: find N weights λ_α summing to 1 and minimizing $Var[\hat{Z} - Z_0]$. This is classically solved by the method of Lagrange multipliers. We consider the function

$$Q = Var[\hat{Z} - Z_0] + 2\mu\left(\sum_\alpha \lambda_\alpha - 1\right)$$

where μ is an additional unknown, the Lagrange multiplier, and determine the unconstrained minimum of Q by equating its partial derivatives to zero:

$$\frac{\partial Q}{\partial \lambda_\alpha} = 2\sum_\beta \lambda_\beta \sigma_{\alpha\beta} - 2\sigma_{\alpha 0} + 2\mu = 0, \qquad \alpha = 1, \ldots, N$$

$$\frac{\partial Q}{\partial \mu} = 2\Big(\sum_\alpha \lambda_\alpha - 1\Big) = 0$$

(That the extremum is indeed a minimum is again guaranteed by the convexity of $Var[\hat{Z} - Z_0]$ as a function of the λ_α). This leads to the following set of $N+1$ linear equations with $N+1$ unknowns:

$$\begin{cases} \sum_\beta \lambda_\beta \sigma_{\alpha\beta} + \mu = \sigma_{\alpha 0}, & \alpha = 1, \ldots, N \\ \sum_\alpha \lambda_\alpha = 1 \end{cases} \tag{1.12}$$

This is the Ordinary Kriging System.

The kriging variance is obtained by premultiplying the first N equations of (1.13) by λ_α, summing over α, and then using the last equation. The result is the OK variance:

$$\sigma^2_{OK} = E[\hat{Z} - Z_0]^2 = \sigma_{00} - \sum_\alpha \lambda_\alpha \sigma_{\alpha 0} - \mu$$

The kriging variance provides a measure of the error associated with the kriging estimator. Notice that it does not depend on the values of the data but only on their locations.

The linear system (1.13) has a unique solution if and only if the covariance matrix $\Sigma = [\sigma_{\alpha\beta}]$ is strictly positive definite, which is the case if we use a strictly positive definite covariance function model and if all data points are distinct.

The condition that the kriging weights add up to 1 entails that the kriging error $(\hat{Z} - Z_0)$ is an allowable linear combination and therefore its variance can be calculated with the variogram, substituting $-\gamma$ for σ in the Ordinary Kriging System:

$$\begin{cases} \sum_\beta \lambda_\beta \gamma_{\alpha\beta} - \mu = \gamma_{\alpha 0}, & \alpha = 1, \ldots, N \\ \sum_\alpha \lambda_\alpha = 1 \end{cases} \tag{1.13}$$

and the OK variance becomes:

$$\sigma^2_{OK} = \sum_\alpha \lambda_\alpha \gamma_{\alpha 0} - \mu$$

Finally, it is important to note that:

- The kriging estimator is an exact interpolant. If x_0 coincides with a sample point, say x_1, then $\hat{Z}$ is equal to $Z(x_1)$. The $\hat{Z} = Z(x_1)$ is certainly the best

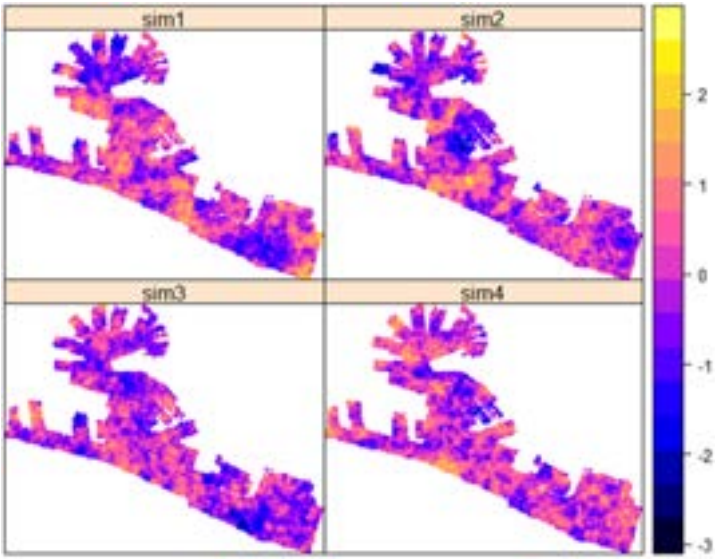


Figure 1.7: Four simulations of an environmental variable using sequential Gaussian simulations.

estimator of $Z(x_1)$ in the m.s.e. sense as it makes the error exactly zero. The kriging variance σ^2_{OK} is naturally also zero.

- Since kriging performs a linear averaging, we expect kriging estimates to be less dispersed than the data (smoothing effect) [Deutsch et al., 1992].

1.4 Sequential Simulations

In the previous section it was anticipated that the kriging method tends to underestimate values that are larger than average and to overestimate those that are smaller, since kriging performs a linear averaging, its estimates are less dispersed than the data. This is called smoothing effect.

Although a kriged map shows the best (i.e. minimum variance) estimates of Z, it does not represent the variations well; this loss of information and detail in the variation could be misleading. To obtain a result that retains the possible variation of the environmental variable, another approach can be used: simulations.

Clearly a simulation is not reality but only a possible version of it (see Figure 1.7), among myriads of others. Using a simulation approach, it is possible to obtain many equally probable realizations that are as likely as the reality and have the same statistical characteristics. In this way dense fields of values from sparse data are obtained, just as we do by kriging, but the variance in the original data is retained. In addition, the simulation approach also allows to obtain a measure of the uncertainty of the estimates that could be used to evaluate and, possibly,

improve the performance of the estimation process.

The sequential approach is the most straightforward method for simulating a RF. Each value is simulated sequentially according to its conditional cumulative distribution function, which must be determined at each location to be simulated. The conditioning data comprise all the original data and all previously simulated values within the neighbourhood of the point being simulated.

The aim of sequential simulation is to obtain an estimation of a RF $\{Z(x), x \in D \subset \mathbb{R}^d\}$, where the domain D is discretised with M points. So, consider a vector-valued random variable $Z = (Z(x_1), Z(x_2), \ldots, Z(x_M))$ for which a realization of the sub-vector $(Z(x_1), Z(x_2), \ldots, Z(x_N))$ is known and equal to $(z(x_1), z(x_2), \ldots, z(x_N))$. The distribution of the vector Z conditional on $Z(x_i) = z(x_i)$, $i = 1, 2, \ldots, N$ can be factorized in the form:

$$
\begin{aligned}
&P\Big[z(x_{N+1}) \leq Z(x_{N+1}) < z(x_{N+1}) + dz(x_{N+1}), \ldots, \\
&\quad z(x_M) \leq Z(x_M) < z(x_M) + dz(x_M) | z(x_1), \ldots, z(x_N)\Big] = \\
&= P\Big[z(x_{N+1}) \leq Z(x_{N+1}) < z(x_{N+1}) + dz(x_{N+1}) | z(x_1), \ldots, z(x_N)\Big] \\
&\times P\Big[z(x_{N+2}) \leq Z(x_{N+2}) < z(x_{N+2}) + dz(x_{N+2}) | z(x_1), \ldots, z(x_N), \\
&z(x_{N+1}) \leq Z(x_{N+1}) < z(x_{N+1}) + dz(x_{N+1})\Big] \\
&\times \ldots \\
&\times P\Big[z(x_M) \leq Z(x_M) < z(x_M) + dz(x_M) | z(x_1), \ldots, z(x_N), z(x_{N+1}), \ldots, \\
&z(x_{M-1}) \leq Z(x_{M-1}) < z(x_{M-1}) + dz(x_{M-1})\Big]
\end{aligned}
\tag{1.14}
$$

Therefore the vector Z can be simulated sequentially by randomly selecting $Z(x_i)$ from the conditional distribution $P[Z(x_i) < z(x_i) | z(x_1), \ldots, z(x_{i-1})]$ for $i = N + 1, \ldots, M$ and including the outcome $z(x_i)$ in the conditioning data set for the next step.

The practical difficulty is that in general, the conditional probabilities involved in Formula (1.14) can not be calculated, except in the ideal case of a Gaussian RF. Therefore, the application of the above method to the simulation of a Gaussian RF, $\{Y(x), x \in D \subset \mathbb{R}^d\}$, known as Sequential Gaussian Simulation (SGS), is straightforward. Indeed for a Gaussian RF with known mean, the conditional distribution of $Z(x_i)$ is Gaussian, with mean $\hat{Z}(x_i)$ and variance $\hat{\sigma}^2_K(x_i)$, where $\hat{Z}(x_i)$ is the simple kriging estimator of $Z(x_i)$ from $\{Z(x_j), j < i\}$, and $\hat{\sigma}^2_K(x_i)$ the associated kriging

variance. In Section 1.4.2 the steps of SGS to obtain an estimation and a measure of uncertainty of an environmental variable will be shown.

In general, the variable to be simulated is not Gaussian. In the stationary case, SGS is therefore applied after a preliminary Gaussian transformation of the data: the Normal Score Transform.

1.4.1 Normal Score Transform

Some statistical methods require the input data to be normally distributed, as mentioned for the SGS. The normal score transformation (NST) is designed to transform data so that it closely resembles a standard normal distribution (i.e. with zero mean and unit variance).

In geospatial applications, it is rare to encounter a variable with a Gaussian spatial distribution, however it is usually possible at least to transform it into one with a Gaussian marginal. If $Z(\cdot)$ has a continuous marginal distribution $F(z)$ and if $G(y)$ stands for the standard Gaussian distribution, the transformation $Y = G^{-1}(F(Z))$ transforms $Z(\cdot)$ into an SRF $Y(\cdot)$ with standard Gaussian marginal and is called the NST. Conversely, Z can be regarded as the transform of the Gaussian Y by $Z(x) = \phi(Y(x))$, and this will turn out to be the useful formulation. The function $\phi = F^{-1} \circ G$ is called the anamorphosis function. It can be represented by its expansion into Hermite polynomials (Section 2.1.1). The normal score transform is the function ϕ^{-1}.

In practice, NST converts data in normal data by ranking the sampled values from lowest to highest and matching these ranks to equivalent ranks generated from a normal distribution. Steps in the transformation are as follows: data is sorted and ranked, an equivalent rank from a standard normal distribution is found for each rank from data, and the normal distribution values associated with those ranks make up the transformed data. The ranking process can be done using the frequency distribution or the cumulative distribution of the data.

Normal scores of sampled data become the input of the SGS. The output of them will be a set of values associated to each point/node of the grid with standard normal distribution. To back transform these values in order to have them in the original scale, first get the cumulative frequency for a value on standard normal distribution curve, then, go to the same cumulative frequency on original data curve, and read the value corresponding to this cumulative frequency value.

The NST and its back transformation have been implemented in *C++* as described in Section 1.5.2.

1.4.2 Sequential Gaussian Simulations

Several simulations (n_{sim}) are computed to obtain different representations of the same environmental variable varying the order in which the locations are visited during the simulation. At the end of SGS, for each point that discretises the domain, the simulated values are averaged in order to obtain a single value to map. In parallel, also the variance is computed to obtain a measure of uncertainty of the estimates. The mapping on the survey domain of the mean and the variance of estimations of the n_{sim} simulations produces an estimation map and an uncertainty map.

To summarize the steps of SGS:

1. Preprocess known values to guarantee they follow a Gaussian distribution. If not, apply a Normal Score Transformation;
2. Compute and fit the variogram;
3. Specify a grid (M nodes) on which one wants to simulate or a point cloud where to obtain the estimation values;
4. Determine the sequence in which the points in the domain will be visited for each simulation (e.g., randomly).
5. Simulate at each point $x_j \in \mathbb{R}^3$, $j = 1, \ldots, M$:
   A. Use kriging to obtain $\hat{Z}(x_j)$ and $\hat{\sigma}^2_K(x_j)$
   B. Draw a value $\hat{z}(x_j)$ at random from a normal distribution: $N(\hat{Z}(x_j), \hat{\sigma}^2_K(x_j))$
   C. Map point x_j to value $\hat{z}(x_j)$ and add it to the SGS set of known values
   D. Proceed to the next node of the grid/point cloud and repeat steps until all of the M nodes/points have been simulated
6. Back transform the simulated values if necessary.
7. Repeat for n_{sim} simulations.

Improving the performance and robustness of algorithms on new high-performance parallel computing architectures is a key issue in efficiently performing 3D analysis with large amount of data and in the real-time survey where reducing the computational time is a crucial task. In [Nunes and Almeida, 2010] a parallelisation of SGS is proposed and this approach is implemented in *C++* so as to be faster and more efficient in providing estimates even for models with a large number of cells.

1.5 Algorithmics

In this section, the main algorithms for the implementation in *C++* of the theoretical tools mentioned in the previous sections are described. This represents the contribution elaborated during the thesis to the development of a geostatistics toolkit for spatial data analysis.

1.5.1 Automatic Fitting of Variogram

Computing the variogram is an important step of the adaptive sampling strategy for environmental applications. Providing support for the automatic fitting of the "best" solution is important to ensure wider applicability of the methodology proposed, by reducing the need for user interaction. During the thesis work, we have addressed this problem defining a specific function, called `fit_variogram`, which implements the automatic variogram fitting. The function tests several models and parameters and identifies the "best" solution automatically.

First of all, a function to compute the experimental variogram from data has been implemented to use it as the input of the `fit_variogram` function. It is important to point out that data for computing the experimental variogram, and then using the `fit_variogram`, are assumed standard normally distributed.

Since in spatial data analysis the points that are closer (low values of lag h) are usually more important than those that are further away (high values of lag h), an alternative version of the traditional variogram estimation has been implemented. So, it is better to be more precise in modeling the experimental variogram for smaller distances rather than for large h values. Furthermore, the greater the number of points to interpolate of the experimental variogram, the better the accuracy of the fit of the interpolating function. Therefore, the number of points where to calculate the experimental variogram at the distances of greatest interest (small h) has been enhanced and on the other hand, it has been made coarser for those of least interest (large h). This is an innovative technique to compute the experimental variogram and it will provide a better fit of points at small lags. For this the function `experimental_variogram_with_variable_lag` has been implemented and Figure (1.8) shows an example of an experimental variogram calculated with variable lags as h increases.

In the function `experimental_variogram_with_variable_lag` the user can also define the parameter δ (the default value is 15) to specify the number of intervals into which to split the range.

For now the function `experimental_variogram_with_variable_lag` only man-

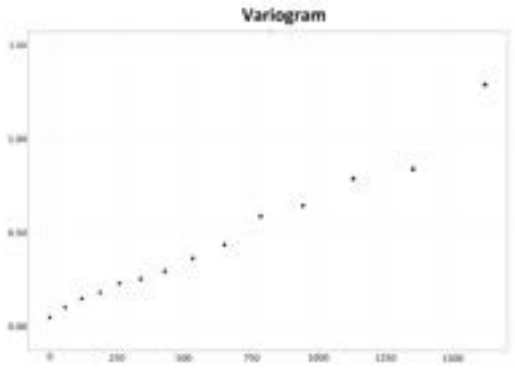


Figure 1.8: Experimental variogram with variable lags. The width of the intervals grows linearly with h.

ages isotropic variograms. In the future this will be extended also for anisotropic spatial distributions, namely when the variogram varies with directions, and an algorithm will be implemented that calculates directional variograms and models them.

Once the experimental variogram is estimated, the function `fit_variogram` can interpolate the values of $\gamma(h)$ with a function in order to have a measure of variogram for any distance.

At the state of the art, several proposals exist for the fitting of the variogram. As reference of our work, [Cressie, 1985] uses the Weighted Least Squares (WLS) to select the best model that interpolates the experimental variogram. The method of WLS is shown to be an appropriate way of fitting variogram models. The weighting scheme automatically gives most weight to early lags and down-weights those lags with a small number of pairs.

At first the function `fit_variogram` computes the *sill* and *nugget* parameters. Given the assumption that the input data is standard normally distributed, the sum of *sill* and *nugget* is 1. The *nugget* is computed considering samples at short distance and computing their $\gamma(h)$ and the *sill* is equal to $(1 - nugget)$. Now, the function `fit_variogram` tests three types of models: Spherical (Sph), Exponential (Exp) and Gaussian (Gau). These are the most frequent models in geostatistics and they are defined in Section 1.2.2. In the future the number of models to be tested may be increased. For each of the three models (Sph, Exp, Gau), several values of *range* is tested to select the "best" function to interpolate points of the experimental variogram. The *range* values are made to vary between 0 and the maximum sampled distance (max_{dist}) and, each time, increased by a quantity that depends on the user (`range_precision`). Weighted Least Square (WLS) is used as the selection criterion: once a type of model ($m = \{Sph, Exp, Gau\}$) and a value of

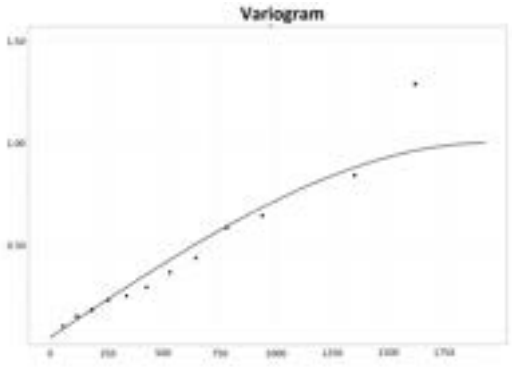


Figure 1.9: Experimental variogram of Figure 1.8 fitted with a Spherical model with range = 1950 meters, sill = 0.95 and nugget = 0.05.

range ($r \in (0, max_{dist})$) are fixed, it is possible to compare the theoretical function defined with these parameters ($\tilde{\gamma}_{m,r}(h)$) respect to the value of the experimental variogram ($\gamma(h)$) at the same value h:

$$WLS_{m,r} = \sum_{i=1}^{\delta} \Big(\gamma(h_i) - \tilde{\gamma}_{m,r}(h_i)\Big)^2 * w_i$$

where $w_i = \frac{N_i}{h_i}$ is the weight and N_i is the number of pairs used to estimate the value $\gamma(h_i)$ at approximately distance h_i. Therefore, the weight w_i is directly proportional to N_i and inversely proportional to the distance (h_i) between the pairs of points. In this way more importance is given to interpolate well the values of the experimental variogram with smaller lag and with a relevant number of pairs to estimate them.

It is important to note that since the distribution of the variable is supposed to be Gaussian (or at least becomes Gaussian through a transformation) only the points with $\gamma(h_i) < 1$ are considered in the interpolation process.

After calculating the WLS for all three types of models and for several values of the range, the combination with the lowest WLS is chosen. Figure 1.9 shows the fitting of the experimental variogram with a Spherical model with *range* of 1950 meters, *sill* equal to 0.95 and *nugget* equal to = 0.05.

Our procedure of automatic fitting is able to work also when the degree of variability between sample pairs at short distance lags is high. An example of experimental variogram with high value of *nugget* is shown in Figure 1.10. This is the same fitted variogram in [Cressie, 1985]. Comparing the results, the values of the *range* are very closer, but our function selects the Gaussian model to fit the points of the experimental variogram, instead of the spherical model of the method proposed in [Cressie, 1985]. This difference between the selected model types is due to the fact that our function considers in the interpolation only the points that

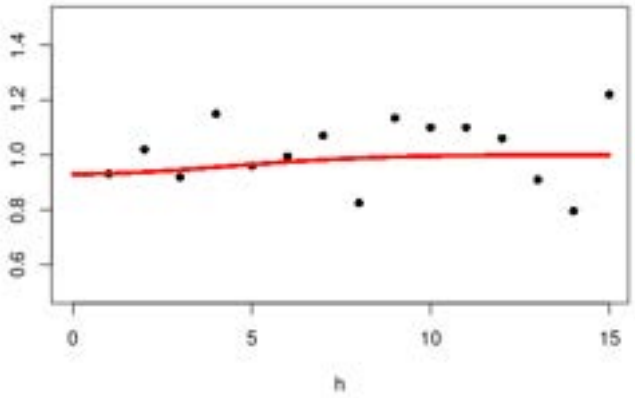


Figure 1.10: Fitting of the experimental variogram as in [Cressie, 1985] (figure 1(a)). Automatic fitting estimates a Gaussian model with parameters: $sill = 0.07$, $nugget = 0.93$ and $range = 5.9$.

have a $\gamma(h)$ value less than one. For a numerical comparison, the Mean Square Error [Hastie et al., 2009] is computed for both models: the value for the Gaussian model is 0.028, while the value for the spherical model is 0.043. Therefore, our method interpolates points with ordinate less than one better than the method in [Cressie, 1985].

Another interesting function that we have developed is the computation of the variogram considering the geodesic distance instead of the Euclidean distance. The motivation of this implementation is related to the framework of applications of geospatial analyses. Indeed, very often the area of interest has a very complex shape and the distance between two points, that would be very close in terms of Euclidean distance, could be much greater. In general, a geodesic is commonly a curve representing the shortest path between two points in a surface. This distance respects the constraints of the geometric model with which the area of interest of the survey is defined. It represents the shortest path that a person would have to physically tread in order to connect two points in space. At the moment this function is not integrated into the real time system due to the computational cost that involves the calculation of the geodesic distances. In the future, thanks to new and ever faster approaches and to parallel computing, it will be possible to apply the geodesic variogram in real time sampling strategies.

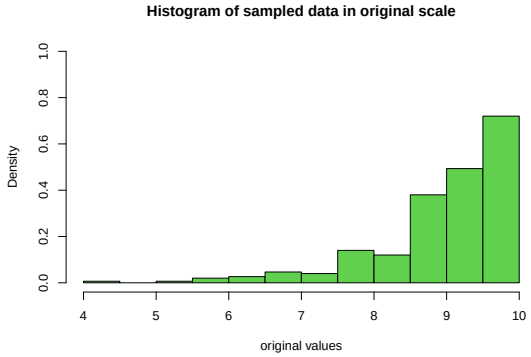


Figure 1.11: Sampled data in the original scale of a random variable with a logaritmic trend.

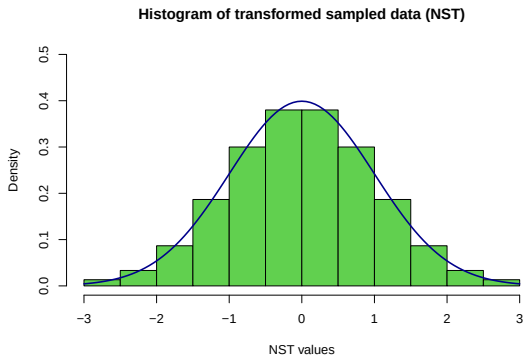


Figure 1.12: Sampled data after the NST. The blue line represents the standard normal distribution.

1.5.2 Normal Score Algorithm

We have implemented in *C++* the function able to convert any distribution of data in a standard normal distribution and also the function that, after the estimation procedure (kriging or SGS), back-transforms the normal values into the original scale.

The first function is called `normal_score` and takes in input a vector of sampled values and returns an object of type `normalscore` where the first component, `x`, of this object corresponds to the sorted original values, the second component, `nsco`, is the vector of sorted standard normal values from the NST and the third component, `values`, is the vector of standard normal values ordered respect to the input data. The latter is used as input in the SGS. In Figure 1.12 and 1.11 there is an example of a normal score transform of data with a logaritmic trend.

The procedure of back-transform can be carried out using two different strategies depending on the type of extrapolation the user selects. Indeed, it is possible that SGS provides extreme values out of the range (maximum - minimum) of original data. In these cases it is possible to choose if the back-transform will revert more extreme score values to the original minimum and maximum values or if an extrapolation is needed. The extrapolation is based on standard deviation (`std`) of the initial data. The new minimum and maximum derive from the old ones $\pm$ `std`. However, this operation is quite dangerous because estimations of the environmental variable could assume unacceptable values (out of its domain). For this reason, the implementation needs a subroutine: it checks if the new minimum and maximum derived from extrapolation respect the domain of the variable (`min_value` and `max_value` are two parameters of the function and they define the domain of the

environmental variable; e.g. for Ph the minimum should be `min_value`= 0 and the maximum `max_value`= 14), otherwise (if for example the new minimum with extrapolation is a negative value for Ph) the subroutine sets as new minimum and/or maximum the extremes of the domain. Once the minimum and maximum are computed, a linear interpolation is carried out in order to obtain the score estimated values in original scale.

Chapter 2

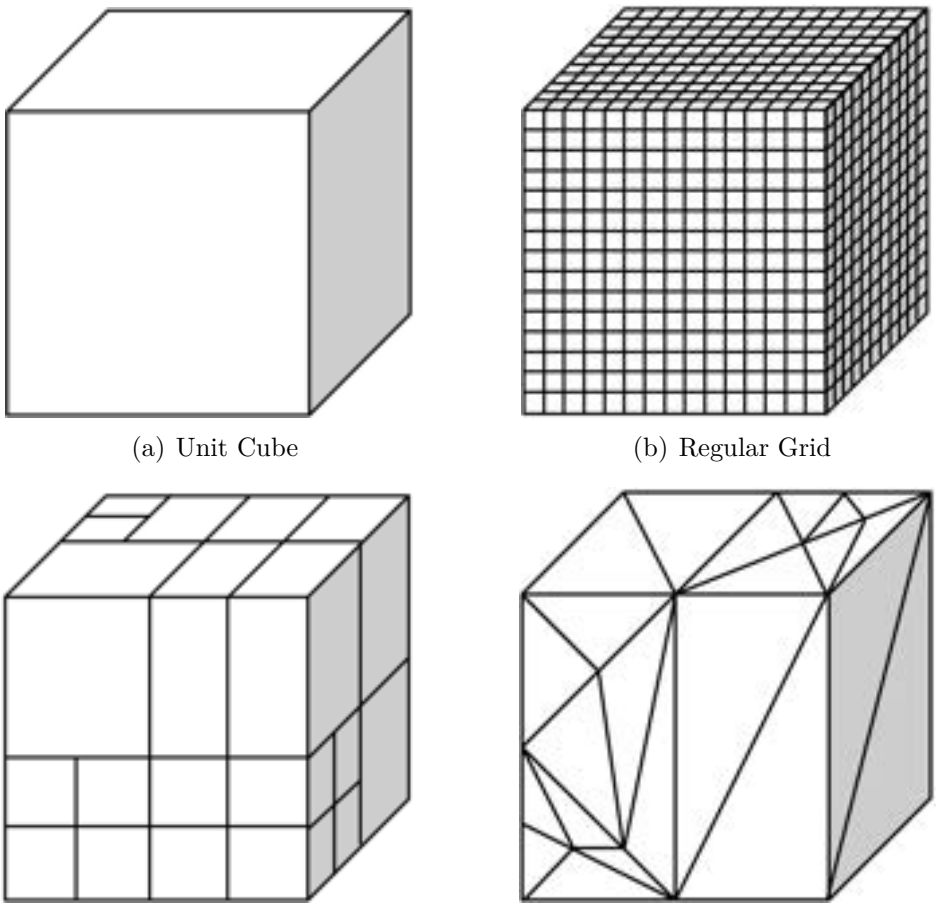
Change of Support Models

Sample support generally refers to the length, area, or volume associated with a measurement [Goovaerts, 2016]. In many applications of spatial data analysis, data are measurements recorded and analysed at distinct points in space (point support). To build the estimate of the spatial variable distribution over an area of interest, however, it is necessary to map the point estimates over an area or a volume. Usually the area of interest is *discretised*, that is, represented by a finite set of cells whose shape, size and spatial structure may vary. These cells are usually named "support", with "block support" used to indicate volumetric cells.

In Figure 2.1 examples of discretisations of a reference unit cubic domain are reported with different supports: in (b) a regular grid is shown, where the support is defined by uniform and cubic blocks, while in (c) and (d) the grid is unstructured with two different shapes of support, respectively, non uniform hexahedral blocks and tetrahedra.

Changing the support of a variable (often by averaging or aggregating) creates a new variable that is related to the original one but it has different statistical and spatial properties. The problem of how the spatial variation of one variable is related to the spatial variation of another variable derived from it but with different supports, is called *change of support* problem [Gotway Crawford and Young, 2005]. The change of support problem lies in predicting the change of distribution when passing from one size of support to another one, generally from a point to a block. Therefore, whenever the support in which the samples are collected is different from the support in which the estimates are required, it is necessary to take into account this mismatch among supports in the process of estimating a stationary random field $\{Z(x), x \in \mathbb{R}^d\}$.

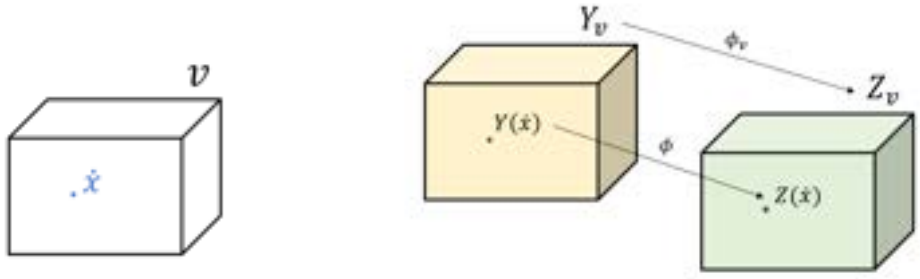
Consider a stationary random field (SRF), $\{Z(x), x \in D \subset \mathbb{R}^d\}$, with marginal distribution function $F(z)$. The aim is to determine the estimates associated with



(a) Unit Cube (b) Regular Grid

(c) Irregular Grid (d) Tetrahedral support

Figure 2.1: Unit cube discretisation using several supports: top left, the reference domain represented by a unit cube; top right, the unit cube discretised with a regular grid where each element has the same size; bottom left, the unit cube discretised with an irregular grid; bottom right, the unit cube discretised with tetrahedra.



(a) (b)

Figure 2.2: (a) A block v ($d = 3$) where $\dot{x}$ is a uniform random point within the block; (b) Two transformation function are defined: ϕ for the point support and ϕ_v for the block support.

the block support v (Figure 2.2(a)); for this, it will suffice to determine the marginal distribution $F_v(z)$ of the regularized SRF $Z_v(x)$. The distribution of block variable $Z_v(x)$ must satisfy three properties:

1. The distributions $F(.)$ and $F_v(.)$ must have the same mean m

2. The variance of the distribution $F_v(.)$ must be equal to the variance given by $\sigma_v^2 = \frac{1}{|v|^2} \int_v \int_v C(x - x')dxdx'$, where $C(h)$ is the covariance of $Z(x)$

3. The distribution $F_v(.)$ must be less selective than $F(.)$. To know whether a given distribution is more selective than another, the theorem established by Cartier [Alfsen, 2012] can be used: let Z_1 and Z_2 be two variables with distributions F_1 and F_2, respectively, the cdf F_1 is more selective than cdf F_2 if and only if there exists a bivariate distribution F_{12} with marginals F_1 and F_2 and such that $\mathbb{E}[Z_1|Z_2] = Z_2$ [Chiles and Delfiner, 2009].

In this chapter, the mathematical models used to sustain the change of support are introduced. In particular, in Section 2.1 the Discrete Gaussian Model is described: this is one of the most used models to change from a puntcual measure to a regular volumetric block. This model is able to deal with structured grid (Figure 2.1(b)) where the sizes of the cells are all the same. When irregular structures of the survey grid are involved (Figure 2.1(c) and 2.1(d)), this definition of change of support models is not enough. For this, in Section 2.2 the discrete Gaussian Model is extended to handle also unstructured grids. Section 2.3 shows how such model are implemented in *C++* code. Finally, in Section 2.4 an experiment is designed to evaluate the effectiveness of these models in the estimation process.

The change of support model discussed in this chapter will be integrated into a new procedure of sampling to improve the results in the estimation process when the support of the sampling domain is volumetric and its blocks have different sizes.

2.1 Discrete Gaussian Model

[Matheron, 1963] proposes the Discrete Gaussian Model (DGM) for RFs Z that are transforms of an SRF Y with a Gaussian marginal distribution.

Consider a block v and a uniform random point $\dot{x}$ within v (Figure 2.2(a)). The distribution of the random variable $Z(\dot{x})$ is the marginal distribution F of the SRF $Z(.)$. Very often, in actual cases, this distribution is not Gaussian. For this, $Z(\dot{x})$ is considered as the transform of a Gaussian of the form

$$Z(\dot{x}) = \phi(Y(\dot{x})) \tag{2.1}$$

where Y is a standard normal variable and $\phi = F_Z^{-1} \circ G$, with G as the standard Gaussian cumulative distribution function (cdf).

Similarly, the mean grade Z_v of the block v is of the form

$$Z_v = \phi_v(Y_v) \tag{2.2}$$

where Y_v is a standard normal variable and ϕ_v the block transformation function (Figure 2.2(b)). Global estimation of the distribution of blocks amounts to determine the block transformation function [Zaytsev, 2016].

The crucial assumption of the DGM is that the bivariate distribution of the $(Y(\dot{x}), Y_v)$ pair is also Gaussian characterised by a correlation coefficient named r.

The block transformation function and its distribution are then derived (see Section 2.1.2), but before explaining this calculation it is necessary to introduce Hermite polynomials.

2.1.1 Hermite Decomposition

This family of polynomials is important because it will help to parameterise conditional distributions. Hermite polynomials are defined by Rodrigues' formula:

$$H_n(y) = \frac{1}{\sqrt{n!}g(y)} \frac{d^n g(y)}{dy^n}, \quad \forall n \geq 0 \tag{2.3}$$

where n is the degree of the polynomial, $\sqrt{n!}$ is a normalization factor, y is a Gaussian or normal value, and $g(y)$ is the standard Gaussian probability density function (pdf) defined by $g(y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}}$. For a given value of y, the polynomial of degree n can easily be calculated [Ortiz et al., 2005].

A useful recursive expression exists to calculate polynomials of higher orders:

$$H_{n+1}(y) = -\frac{1}{\sqrt{n+1}} * y * H_n(y) - \sqrt{\frac{n}{n+1}} * H_{n-1}(y) \quad \forall n \geq 1 \tag{2.4}$$

This expression along with the knowledge of the first two polynomials is enough for fast calculation up to any order. With $g(y)$ as the standard Gaussian probability density function, the first three polynomials are:

$$H_0(y) = 1$$

$$H_1(y) = -y$$

$$H_2(y) = \frac{1}{\sqrt{2}}(y^2 - 1)$$

Hermite polynomials have some properties as *i)* mean of $H_n(Y)$ is 0, except for the polynomial of order 0, which has a mean of 1; *ii)* variance of $H_n(Y)$ is 1, except again for the polynomial of order 0 which is constant and therefore its variance is 0; *iii)* covariance between $H_n(Y)$ and $H_p(Y)$ is 0 if $n \neq p$. This property is known as orthogonality. Of course, if $n = p$ the covariance becomes the variance of $H_n(Y)$. If all covariances are zero, this is sufficient for full independence if the multivariate distribution is Gaussian.

In summary, Hermite polynomials form an orthonormal basis with respect to the standard normal distribution.

Bivariate Gaussian Distribution

Consider the environmental variable Y distributed in a domain D in the 3D space. Let define the random function model $\{Y(u), \forall u \in D\}$, where u is a location vector in the three-dimensional space. Taking a pair of random variables $Y(u)$ and $Y(u+h)$, they are bivariate Gaussian if their joint distribution is normal with mean vector μ and variance-covariance matrix Σ:

$$\Big(Y(u), Y(u+h)\Big) \sim N_2\Big(\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Sigma = \begin{pmatrix} 1 & \rho(h) \\ \rho(h) & 1 \end{pmatrix}\Big) \tag{2.5}$$

The correlation function $\rho(h)$ gives all the structural information of the bivariate

relationship. Under this assumption, the covariance between polynomials of different order is always 0, and if the order is the same, it identifies the correlation raised to the polynomial's degree power, that is:

$$Cov\Big(H_n(Y(u)), H_p(Y(u+h))\Big) = \begin{cases} (\rho(h))^n & \text{if} \quad n = p \\ 0 & \text{if} \quad n \neq p \end{cases} \tag{2.6}$$

The only term that is left is the covariance between polynomial values of the same degree for locations separated by a vector h and there is no spatial correlation between polynomials of different orders.

Fitting a Function

Any function with finite variance can be fitted by an infinite expansion of Hermite polynomials. The idea is to express the function of $Y(u)$ as an infinite sum of weighted polynomial values:

$$f(Y(u)) = \sum_{n=0}^{\infty} f_n * H_n(Y(u)) \tag{2.7}$$

In order to find the coefficients $f_n, \forall n$, the expected value of the product of the function and the polynomial of degree n is calculated and the result is:

$$E\Big[f(Y(u)) * H_n(Y(u))\Big] = f_n \tag{2.8}$$

It is worth noting that the coefficient of 0 degree corresponds to the mean of the function of the random variable since $H_0(Y(u)) = 1$:

$$f_0 = E\Big[f(Y(u)) * H_0(Y(u))\Big] = E\Big[f(Y(u)) * 1\Big] = E\Big[f(Y(u))\Big] \tag{2.9}$$

The variance of the function of $Y(u)$ can also be calculated and corresponds to the infinite sum of squared coefficients:

$$Var\Big[f(Y(u))\Big] = \sum_{n=1}^{\infty} (f_n)^2 \tag{2.10}$$

In a practical implementation, the infinite expansion of Hermite polynomial is truncated at a given degree P. The truncation could cause a generation of values outside the range of the data. These values can simply be reset to a minimum or maximum values. If the number of polynomials used is large enough, these problems are of limited impact [Ortiz et al., 2005].

Fitting Normal Score Transform

Consider Y as the normal score transform (also known as the Gaussian anamorphosis) of a variable Z with N available samples at locations $u_\alpha, \alpha = 1, \ldots, N$. The cumulative distribution function (cdf) of Z is denoted $F_Z(z)$:

$$y = G^{-1}F_Z(z) \qquad \text{and} \qquad F_Z^{-1}(G(y)) = \phi(y) \tag{2.11}$$

where $\phi = F_Z^{-1} \circ G$ is the anamorphosis function. This normal score transformation relates the sample data $z(u_\alpha)$ with corresponding quantiles of the standard normal distribution $G(y)$. It is possible to parametrise this relationship with a finite number of coefficients using the Hermitian expansion. A Hermite polynomial up to a degree P is used to give an approximation to the shape of the function ϕ:

$$Z(u) = \phi(Y(u)) \approx \sum_{p=0}^{P} \phi_p * H_p(Y(u)) \tag{2.12}$$

In order to calculate the coefficients of this expansion, the following formulas can be used:

$$\begin{cases} \phi_0 = E\Big[\phi(Y(u))\Big] = E\Big[Z(u)\Big] \\ \phi_p = \sum_{\alpha=2}^{N}\Big(z(u_{\alpha-1}) - z(u_\alpha)\Big)\frac{1}{\sqrt{p}}H_{p-1}(y(u_\alpha))g(y(u_\alpha)) \end{cases} \tag{2.13}$$

2.1.2 Block Transformation Function

Exploiting the property (3) introduced at the beginning of this chapter and the Cartier's relation [Chiles and Delfiner, 2009], the block transformation function can be expressed as follow:

$$\mathbb{E}[\phi(Y(\dot{x}))|Y_v] = \phi_v(Y_v) \tag{2.14}$$

Since the $(Y(\dot{x}), Y_v)$ pair is Gaussian, the conditional distribution of Y given $Y_v = y_v$ is Gaussian with mean ry_v and variance $1 - r^2$ [Hastie et al., 2009]:

$$\phi_v(Y_v) = \int \phi(ry_v + \sqrt{1 - r^2}u)g(u)du \tag{2.15}$$

where $g(u)$ is the standard Gaussian probability density function.

In usual cases the actual calculation of $\phi_v()$ is carried out using the expansions of the transformation functions into normalized Hermite polynomials

$$\phi_v(y) = \sum_{p=0}^{\infty} \phi_{vp}H_p(y(u)) \tag{2.16}$$

The coefficients $\phi_p, p = 1, \ldots, P$, calculated in the previous sections, are assumed known and the coefficients ϕ_{vp} are to be determined.

Using the property of Hermite polinomials (Formula (2.6)) and applying relation $\mathbb{E}[H_p(Y(\dot{x})|Y_v)] = r^p H_p(Y_v)$ [Chiles and Delfiner, 2009], the Cartier's relation (Formula (2.14)) is expanded into:

$$\begin{aligned}\mathbb{E}\Big[\phi(Y(\dot{x})|Y_v)\Big] &= \mathbb{E}\Big[\sum_{p=0}^{\infty}\phi_p H_p(Y(\dot{x}))|Y_v)\Big] = \\ &\sum_{p=0}^{\infty}\phi_p \mathbb{E}\Big[H_p(Y(\dot{x}))|Y_v\Big] = \\ &\sum_{p=0}^{\infty}\phi_p r^p H_p(Y_v) = \phi_v(Y_v) = \sum_{p=0}^{\infty}\phi_{vp}H_p(Y_v)\end{aligned} \tag{2.17}$$

Since $H_p(Y_v)$ constitute an orthonormal basis, then $\phi_{vp} = \phi_p r^p$ for all $p = 0, 1, 2, \ldots$. In conclusion,

$$\phi_v(y) = \sum_{p=0}^{\infty}\phi_p r^p H_p(y(u)) \tag{2.18}$$

The correlation coefficient r is selected so that the distribution defined by ϕ_v has the variance given by

$$\sigma_v^2 = \frac{1}{|v|^2}\int_v\int_v C(x - x^{'})dxdx^{'} \tag{2.19}$$

respecting the second property of the distribution of block grades $Z_v(x)$. The variance of $\phi_v(Y_v)$, taken as a function of r, is

$$Var\Big[\phi_v(Y_v)\Big] = \sum_{p=0}^{\infty}(\phi_p r^p)^2 \tag{2.20}$$

and r is the solution of this equation. Variance of $\phi_v(Y_v)$ increases from 0 (for $r = 0$ and assuming that 0^0 is zero) to σ^2 (for $r = 1$), where $\sigma^2 = C(0)$ is the variance of $Z(x)$. Since the value of σ_v^2 is itself comprised between 0 and σ^2, this equation has indeed one and only one solution.

2.2 Discrete Gaussian Model for Unstructured Grids

2.2.1 Unstructured Grids

Structured grids lack the ability to model complex environmental geometries. Especially in environmental practice, it is necessary to give more complex representations of the domain to model these particular geometries. In environmental survey it has become frequent to have a great variety of grid cells of different size and shape to

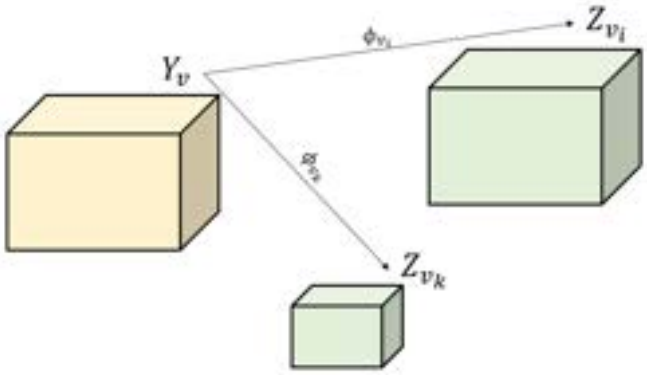


Figure 2.3: Different block transformation functions for different block sizes: v_i and v_k.

enable fine-scale modeling close to important locations and coarse modeling in less important regions. Unstructured grids are introduced in this framework to overcome limitations of structured grids.

Definition 2. An unstructured grid D is composed by a finite number N_b of non overlapping blocks of various sizes:

$$D = \bigcup_{i=1}^{N_b} v_i, \qquad v_i \cap v_k = \emptyset, \forall i \neq k \tag{2.21}$$

and the volumes of the blocks, $\{|v_i|, i = 1, \ldots, N_b\}$, may be different from each other.

2.2.2 DGM for Unstructured Grids

Let $Z(x)$ be an SRF that can be expressed as the transform of an SRF $Y(x)$ with standard normal marginal distribution, $Z(x) = \phi(Y(x))$. Consider a domain D described by an unstructured grid. Consider $v_i \in D$ and let $\dot{x}$ be a uniform random point within v_i, such that $Z(\dot{x}) = \phi(Y(\dot{x}))$. The mean grade $Z(v_i)$ of the block v_i is of the form

$$Z_{v_i} = \phi_{v_i}(Y_{v_i}) \tag{2.22}$$

where Y_{v_i} is a standard normal variable and ϕ_{v_i} is the block transformation function of v_i (Figure 2.3). For each element of the unstructured grid the block transformation function must be determined [Zaytsev, 2016].

The crucial assumption of the DGM becomes that the bivariate distribution of the $(Y_{v_i}, Y(\dot{x}))$ pair is Gaussian with a correlation coefficient r_i.

Moreover, Cartier's relation (2.14) becomes:

$$\mathbb{E}[\phi(Y(\dot{x}))|Y_{v_i}] = \phi_v(Y_{v_i}) \tag{2.23}$$

Block Transformation Function for each v_i

As showed in Section 2.1.2, the calculation of the block transformation functions, $\{\phi_v(Y_{v_i}), i = 1, \ldots, N_b\}$, will be carried out using Hermite polynomials.

For each block v_i , equation (2.16) becomes:

$$\phi_{v_i}(y) = \sum_{p=0}^{\infty} \phi_{v_i p} H_p(y) \tag{2.24}$$

The coefficients $\phi_{v_i p}$ are to be determined.

Using the property of Hermite polynomials (Formula (2.6)) and applying the following relation

$$\mathbb{E}[H_p(Y(\dot{x})|Y_{v_i})] = r_i^p H_p(Y_{v_i}) \tag{2.25}$$

the Cartier's relation (Formula (2.23)) is expanded into:

$$\begin{aligned} \mathbb{E}\Big[\phi(Y(\dot{x})|Y_{v_i})\Big] &= \phi_v(Y_{v_i}) \\ \mathbb{E}\Big[\sum_{p=0}^{\infty} \phi_p H_p(Y(\dot{x}))|Y_{v_i})\Big] &= \phi_v(Y_{v_i}) \\ \sum_{p=0}^{\infty} \phi_p \mathbb{E}\Big[H_p(Y(\dot{x}))|Y_{v_i}\Big] &= \phi_v(Y_{v_i}) \\ \sum_{p=0}^{\infty} \phi_p r_i^p H_p(Y_{v_i}) &= \sum_{p=0}^{\infty} \phi_{vp} H_p(Y_v) \end{aligned} \tag{2.26}$$

Since $H_p(Y_{v_i})$ constitute an orthonormal basis, then $\phi_{v_i p} = \phi_p r_i^p$ for all $p = 0, 1, 2, \ldots$. In conclusion, the result is the following:

$$\phi_{v_i}(y) = \sum_{p=0}^{\infty} \phi_p r_i^p H_p(y(u)) \tag{2.27}$$

Change of Support Coefficient for each v_i

The correlation coefficient r_i is selected for each block v_i so that the distribution defined by ϕ_{v_i} has the variance given by

$$\sigma_{v_i}^2 = \frac{1}{|v_i|^2} \int_{v_i} \int_{v_i} C(x - x^{'}) dx dx^{'} \tag{2.28}$$

The variance of $\phi_v(Y_{v_i})$, taken as a function of r_i, is

$$Var\Big[\phi_v(Y_{v_i})\Big] = \sum_{p=0}^{\infty}(\phi_p r_i^p)^2 \tag{2.29}$$

and r_i is the solution of

$$\sum_{p=0}^{\infty}(\phi_p r_i^p)^2 = \frac{1}{|v_i|^2}\int_{v_i}\int_{v_i} C(x-x^{'})dxdx^{'} \tag{2.30}$$

2.2.3 DGM-2 for Unstructured Grids

This model is a generalization of another version of the DGM presented in [Emery, 2007] and [Rivoirard, 1994]. At the cost of a further, more restrictive, assumption, it provides a simpler approach for computing the change of support coefficients r_i.

The additional assumption is: *for any block v_i and two independent randomized locations $\dot{x}$ and $\dot{x}^{'}$, the bivariate distribution of $Y(\dot{x})$ and $Y(\dot{x}^{'})$ is Gaussian.*

From this assumption, it can be derived the following relation between Y_{v_i} and $Y(v_i) = \frac{1}{v_i}\int_{v_i} Y(x)dx$ for every block [Chiles and Delfiner, 2009]:

$$Y(v_i) = r_i Y_{v_i} \tag{2.31}$$

which provides a simple formula for computing the change-of-support coefficient r_i [Emery, 2007]. In that case, r_i^2 is the block variance of the SRF $Y(v_i)$:

$$r_i^2 = \frac{1}{|v_i|^2}\int_{v_i}\int_{v_i} \rho(x-x^{'})dxdx^{'} \tag{2.32}$$

where ρ is the correlation of the SRF Y.

Since adding a more restrictive assumption implies that the model is less likely to fit the data, the price to pay is in some cases less accuracy in reproducing the histogram of the simulated property [Chiles and Delfiner, 2009].

2.3 Algorithmic

The implementation of the change of support models in a *C++* code includes several steps. First of all, the definition of Hermite decomposition; then the computation of the change of support coefficients and, finally, the back transformation of the output of SGS. These steps are detailed below.

2.3.1 Hermite polynomials

Two functions have been implemented in order to obtain the Hermite polynomials and their coefficients up to degree P. These are useful to define the Hermite decomposition for parameterizing the transformation of the data distribution in a standard normal distribution. This is necessary to apply SGS and to obtain an estimation of the environmental variable from a set of sampled data.

The function `hermite_polynomial` takes as input the sampled data (vector of N elements) and the parameter P, which defines the degree the polynomials must be calculated. This function returns a vector of $N * P$ elements which contains the polynomials of Hermite computed using Formula (2.4).

The second function, `hermite_coefficients`, allows to compute the coefficients of the Hermite decomposition. This function takes as input the sample data both in original and transformed (with normal score transform (NST), Section 1.4.1) scale and the parameter P to define the maximum degree of the Hermite polynomials. Then, data are sorted and the coefficients for $p = 0, \ldots, P$ are computed using Formula (2.13). For example, the coefficient when $p = 0$ is obtained as the mean of data in the original scale. The function returns a vector of $P + 1$ elements which contains the coefficients $\{\phi_p, p = 0, \ldots, P\}$. These are then used for the back-transformation of data, after the simulation process.

2.3.2 Change of support coefficients

To compute the change of support coefficients for each element of the unstructured grid, two approaches can be adopted: the traditional DGM or the DGM-2 as they are described in Section 2.2.2 and 2.2.3, respectively.

Considering the standard approach of DGM, Equation (2.30) must be solved in order to implement a function for computing change of support coefficients. That equation is a polynomial of degree $2P$ in r_i. In this equation, the Hermite coefficients are known using `hermite_coefficients` function; the block-to-block covariance of the i-th block can be computed approximately and how this process works will be explained in Section 3. Looking for a solution of a polynomial of any degree P is not always possible to obtain an exact solution, but in general it is possible to find an approximation using numerical methods, e.g., Newton's method. In particular, Limited-memory BFGS (L-BFGS) [Liu and Nocedal, 1989], is an optimisation algorithm in the family of quasi-Newton methods. It approximates the Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) using a limited amount of computer memory, providing faster solutions of the polynomial. This approach is

used to obtain the solution for r_i. The pseudocode of this function is reported in Algorithm 11 in Appendix B.

Using the approach of DGM-2 it is possible to simplify the computation of the change of support coefficients using Equation (2.32). In this case it will be sufficient to compute the block-to-block covariance using the approach that will be described in Section 3. The pseudocode of this function is shown in Algorithm 12 in Appendix B.

2.3.3 Back tranformation

Change of support coefficients and the Hermite decomposition are the needed ingredients for back-transforming (Formula (2.27)) the data coming from the SGS into the original scale. This is done by taking into account the size of the M blocks of the unstructured grid for which estimations refer to. The implemented function `back_transform_with_hermite` takes as inputs the coefficients of Hermite, the change of support coeffients and the estimated values from SGS. It returns as output a vector of dimension M with back-transformed values in order to map them on the geometric model of the survey domain. The pseudocode of this function is outlined in Algorithm 13 in Appendix B.

2.4 Experiment

To test the performance of the change of support model implemented and its contribution to the quality of the estimation map, the following experiment is designed.

First of all, let us define a continuous synthetic field on a cubic domain 100x100x100 of arbitrary units, which associates to each point with coordinates (x, y, z) a function's value. In particular, the theoretical function that has been tested is

$$f(x, y, z) = x + y + z + \epsilon, \tag{2.33}$$

where ϵ is a random variable with normal distribution and its parameters, mean and variance, are defined a priori. By increasing the values of the variance of ϵ the spatial distribution becomes more noisy and the estimation of the nugget value higher. Therefore, since one of the aims is to test the performance of the change of support models both with high and low nugget scenarios, then the mean and the variance, (μ, σ^2), are considered equal to $(0, 1)$ and $(0, 50)$ respectively. A graphical representation of these functions is shown in Figure 2.4 and these functions will be used as a synthetic field where to collect data of the simulated sampling in our

experiment.

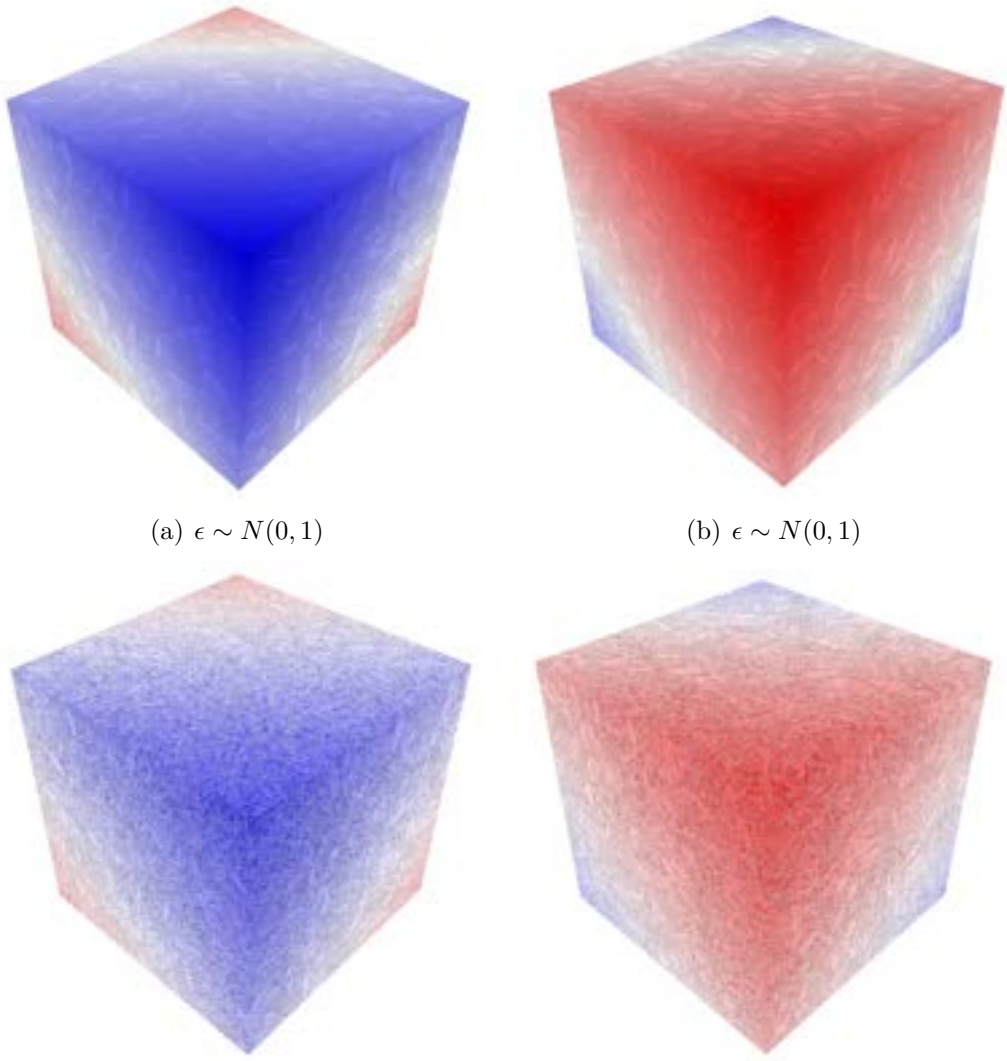
For the visualisation of the estimation map and the evaluation of the different geostatistical methods, the cubic domain has been discretised into several tetrahedral cells of different sizes (unstructured grid) in order to use them as geometric models of the experimental domain. Two geometric models are generated: *(i)* the first is referred as the coarse model (CM) and corresponds to a coarse-scale representation made by $K = 75$ elements; *(ii)* the second is generated starting from the CM by applying several splitting operations (volume, face and edge splits on their middle points) subdividing each tetrahedron into several smaller tetrahedra. This is called fine model (FM) and it has a grid of $M = 172800$ elements ($K << M$). Note that a mapping between the fine and the coarse scale model is defined by grid construction: for each tetrahedron in CM the set of tetrahedra in FM belonging to it is known and determined by a finite and integer number of units. In order to show the effectiveness of the change of support models, the two geometric models have been built with both very small and very large volumes to further emphasize the difference between the sizes of the supports.

In general, the expected behaviour is that larger tetrahedra have larger block-to-block variances and consequently the change of support coefficients will be further away from one. On the other hand, if the volume of tetrahedra is smaller (close to the point-support) the block-to-block variance will be smaller and the change of support coefficients will be close to one.

The change of support models are useless when the volumetric and point support coincide. On the other hand, when the volumes of the blocks are further away from the point support, the estimation without using change of support models (i.e. the assignment of the point estimation value directly to the block) could lead to significant estimation errors.

These considerations also strongly depend on range and nugget values. If the range is much larger than the size of the cell volume, it can happen that even the largest cells of the unstructured grid still have a low block-to-block variance and that their coefficients for the change of support will be very close to one. Vice versa, if the range has a lower value than the size of some cells of the grid, then it is possible that the block-to-block variance values are higher than expected. In general, the value of the change of support coefficient depends on the relationship between the cell size and the range of the spatial distribution. Moreover, when the nugget value is high, also the smaller blocks could have high variance and as consequence the change of support coefficients will be further away from one.

Each element of the CM and FM has been discretised with a finite number



(a) $\epsilon \sim N(0,1)$ (b) $\epsilon \sim N(0,1)$

(c) $\epsilon \sim N(0,50)$ (d) $\epsilon \sim N(0,50)$

Figure 2.4: Synthetic fields of $f(x,y,z) = x + y + z + \epsilon$ on a very fine discretised model, each with two different perspectives. In (a) and (b) $\mu = 0$ $\sigma^2 = 1$; in (c) and (d) $\mu = 0$ $\sigma^2 = 50$.

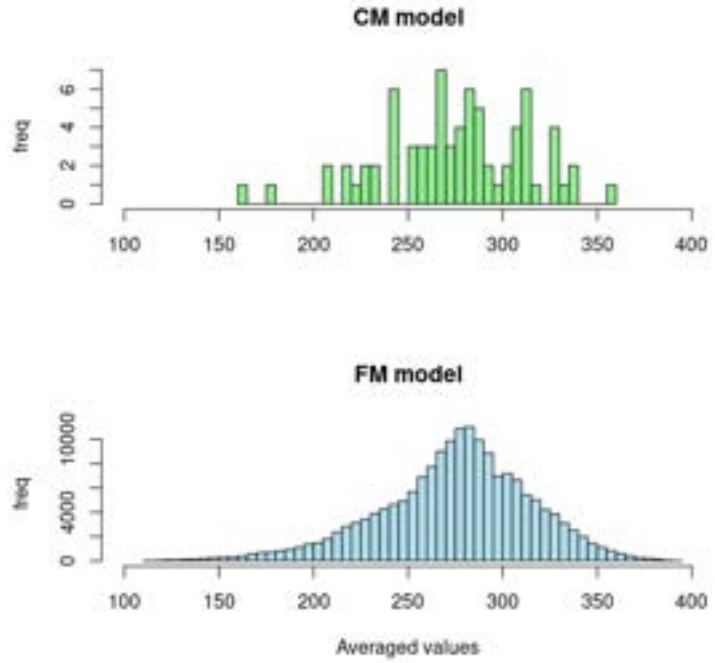


Figure 2.5: Histogram of the averaged values in CM and FM using $\sigma^2 = 1$ for the function $f(x, y, z)$.

of points exploiting the Sobol's algorithm (Appendix A.1). For each point of this sequence, the value of the function of Equation (2.33) is computed. Subsequently, values belonging to the same cell are averaged in order to assign a mean-value of the synthetic field at each tetrahedron of the unstructured grids. The distribution of these values is shown in Figure 2.5 and 2.6. The main statistical information about it is in Table 2.1 and 2.2, respectively, using the variance of ϵ equal to 1 or 50. In this experiment, $N_{seq} = 50$ points are used for the discretisation of each tetrahedron. However, a different number of points could be used for each element of the grid, proportional to its volume: the larger the volume of the tetrahedron, the higher the number of discretisation points.

This averaging procedure will be helpful for the evaluation of the reliability of the results and for the comparison of the estimates using or not the change of support model. The averaged synthetic fields computed with different values of ϵ variance on the CM and FM are shown in Figure 2.8 and 2.7, respectively.

	Min	Median	Mean	Var	Max
CM	162	278.2	274.9	1411.4	355.8
FM	111.4	277.9	274.8	1624.3	392.8

Table 2.1: Main statistical information of the distribution of averaged values in CM and FM with variance of ϵ equal to 1.

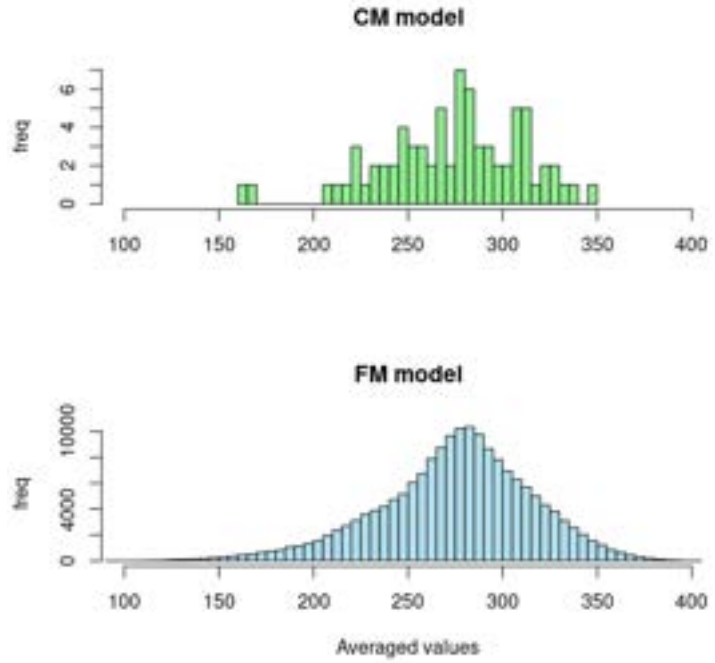


Figure 2.6: Histogram of the averaged values in CM and FM using $\sigma^2 = 50$ for the function $f(x, y, z)$.

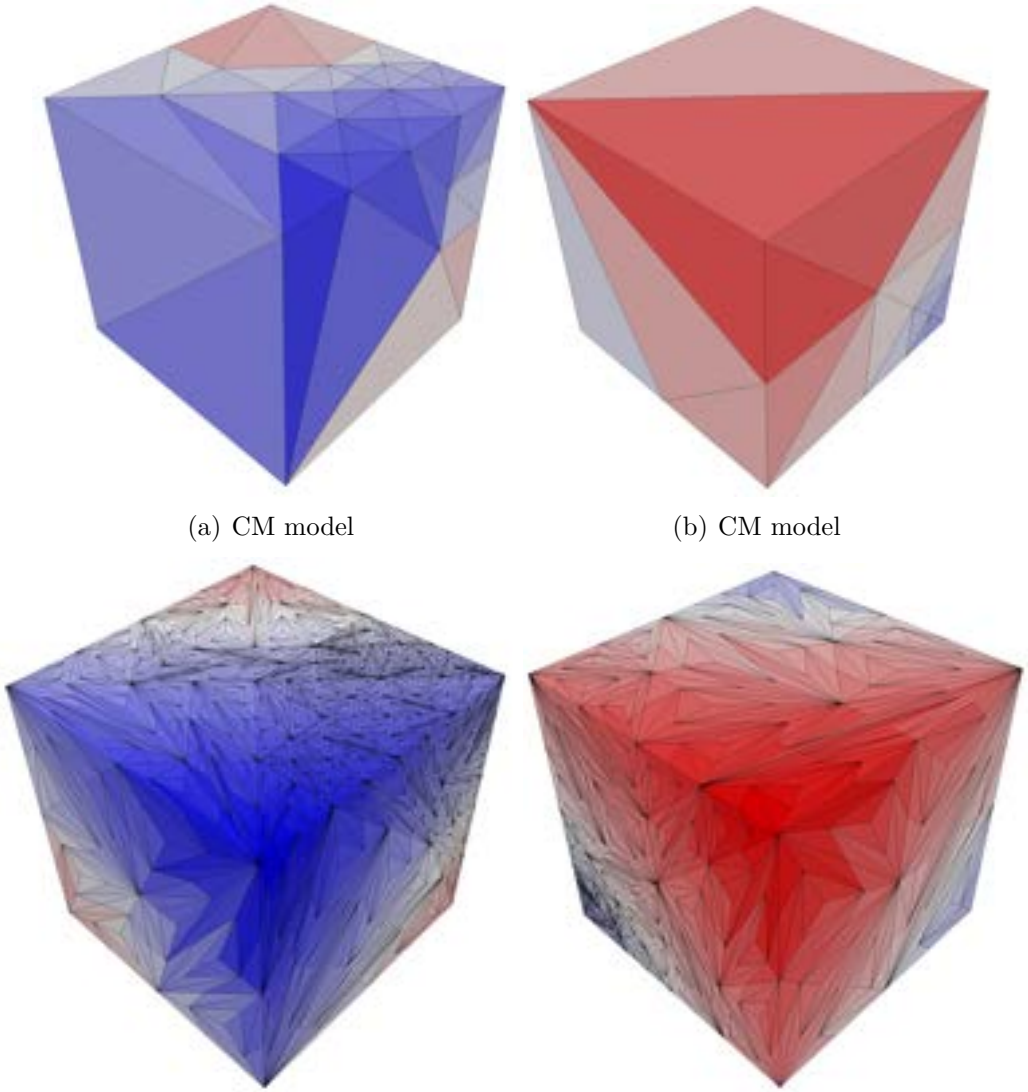
	Min	Median	Mean	Var	Max
CM	162.8	276.8	274.1	1381.615	349.5
FM	93.2	277.8	274.7	1674.849	401.8

Table 2.2: Main statistical information of the distribution of averaged values in CM and FM with variance of ϵ equal to 50.

A set of samples is necessary to apply spatial data analysis and to provide an estimation of the theoretical function over the cubic domain. These samples are randomly selected on the cube and the number of points to consider varies in the experiment in order to evaluate the sensitivity of the estimation process ($N = 40$, $N = 100$ and $N = 200$). For ease of exposition, only graphs and tables with $N = 40$ are depicted (for completeness, some material is reported in Appendix C fot the other values of N). However, at the end of the experiment, considerations are reported for each numerosity.

The plot of sampled data is shown in Figure 2.9. Using these sampled data, the experimental variogram is computed and fitted and the parameters are shown in Table 2.3.

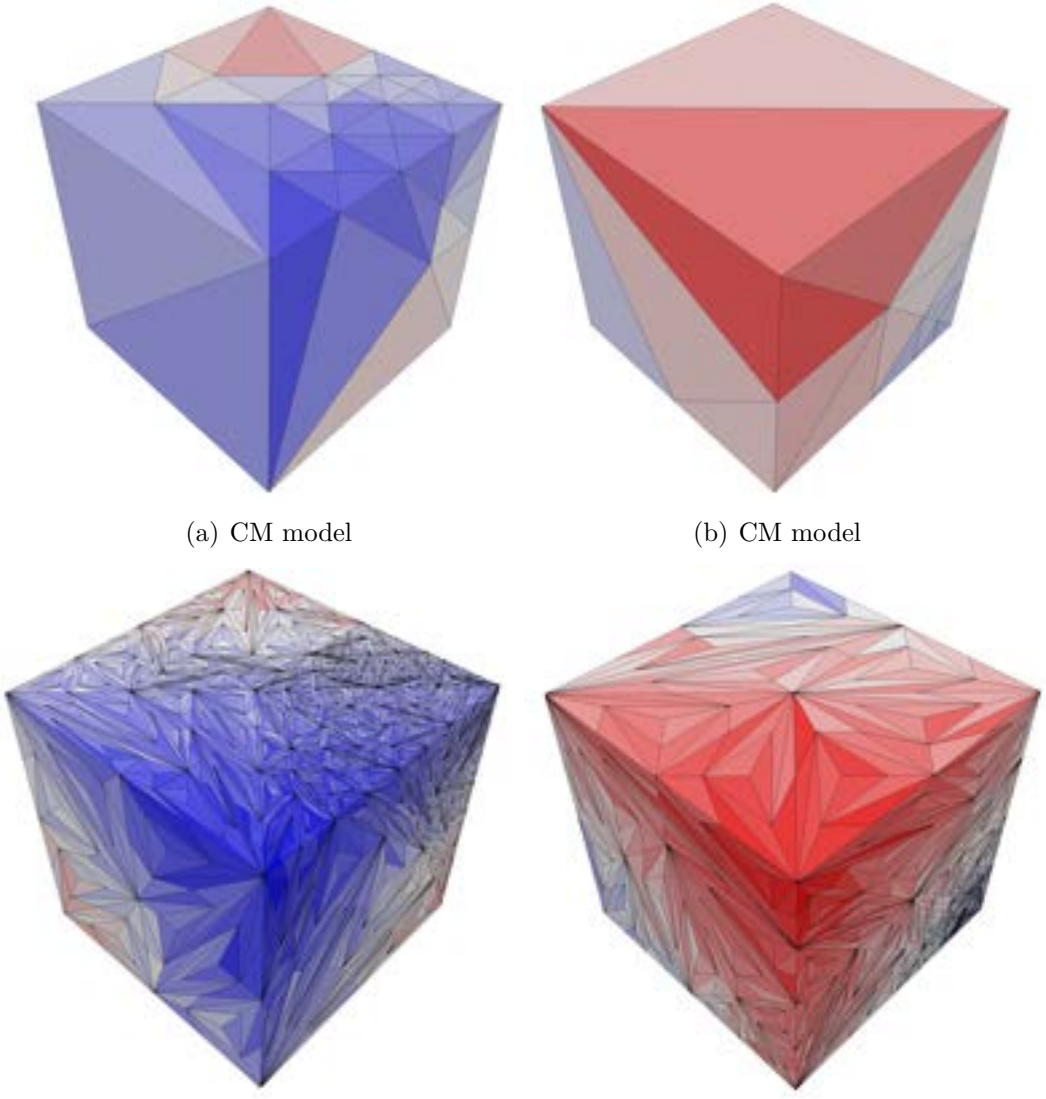
After selecting the number of simulations for the SGS ($n_{sim} = 16$), two procedures are used to obtain the estimation of $f(x, y, z)$: *(i)* the first uses the normal score transform (NST) to back-transform the values of SGS output; *(ii)* the second



(a) CM model (b) CM model

(c) FM model (d) FM model

Figure 2.7: Averaged Synthetic Field with $\mu = 0$ $\sigma^2 = 1$. (a) and (b) represent two different perspectives of the Coarse Model; (c) and (d) two different perspectives of the Fine Model.



(a) CM model (b) CM model

(c) FM model (d) FM model

Figure 2.8: Averaged Synthetic Field with $\mu = 0$ $\sigma^2 = 50$. (a) and (b) represent two different perspectives of the Coarse Model; (c) and (d) two different perspectives of the Fine Model.

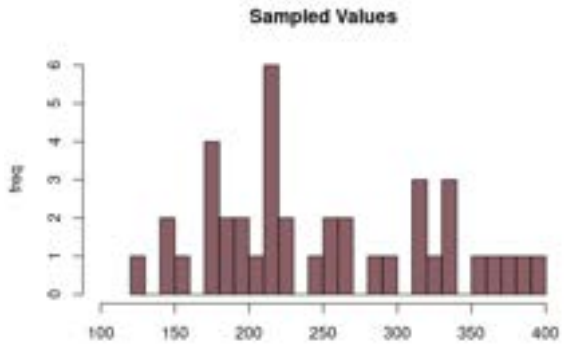


Figure 2.9: Histogram of sampled data ($N = 40$).

	$N = 40$	
	$\sigma^2 = 1$	$\sigma^2 = 50$
Model	Gaussian	Exponential
Nugget	0.002	0.562
Sill	0.998	0.438
Range	61.558	91.792

Table 2.3: Parameters of the fitted variogram with $N = 40$ samples and both values of variance of ϵ: 1 and 50.

uses the DGM model that takes into account the size of the tetrahedra to back-transform the values estimated by the SGS. The distributions of the estimates of the random field with $\sigma^2 = 1$ using both approaches are depicted in Figure 2.10 and 2.11. Results obtained with $\sigma^2 = 50$ are displayed in Figure 2.12 and 2.13.

To give a quantitative measure of the precision of the estimates compared with the averaged synthetic field, both for CM and FM model, the Mean Square Error (MSE) [Hastie et al., 2009] is used. If MSE is smaller using the approach of the DGM, this implies that taking into account the sizes of tetrahedra is effective. Results are summarised in Table 2.4 varying: *(i)* the number of samples used as input of geostatistical methods ($N = 40, 100, 200$), *(ii)* the variance of ϵ ($\sigma^2 = 1, 50$), *(iii)* the use of the DGM or NST and *(iiii)* the coarse or fine model. It is evident from the results that taking into account the volume of the supports and using the DGM improve the estimates of the random field in all the tested conditions, by varying several parameters. The MSE that compares the results with the averaged synthetic field is always lower using the DGM rather than the NST to back-transform the output of the SGS. This occurs both with more or less sampled points, with a high and low variance value (and therefore of the nugget value) and with both coarse and fine model.

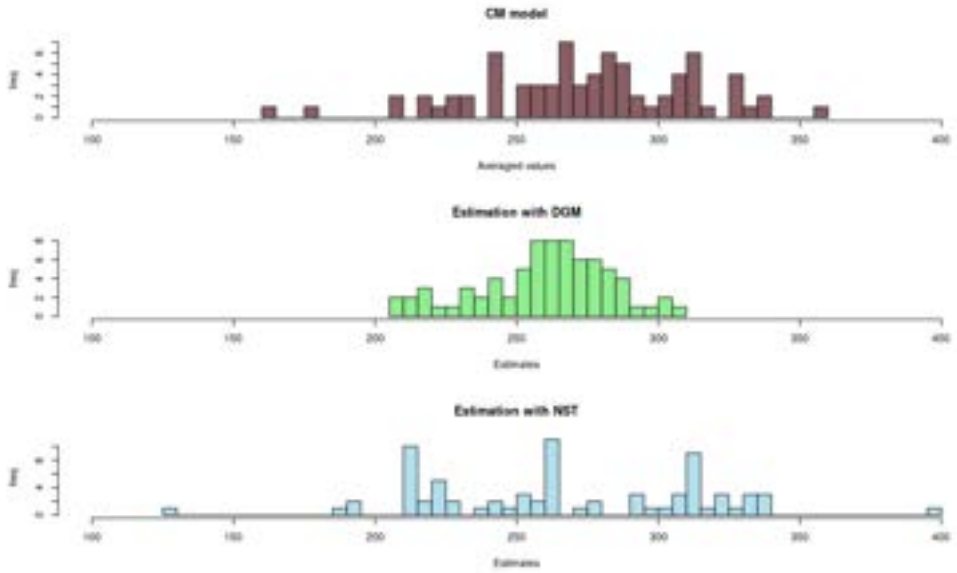


Figure 2.10: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model as in Figure 2.5. The number of sampled points is $N = 40$ and the function in (2.33) has as variance of ϵ equal to one ($\sigma^2 = 1$).

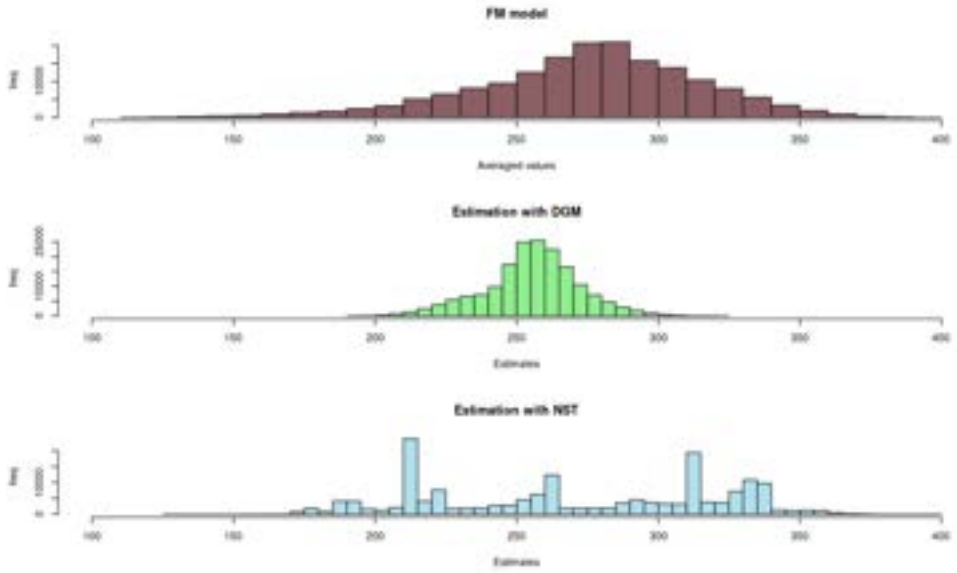


Figure 2.11: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the fine model as in Figure 2.5. The number of sampled points is $N = 40$ and the function in (2.33) has as variance of ϵ equal to one ($\sigma^2 = 1$).

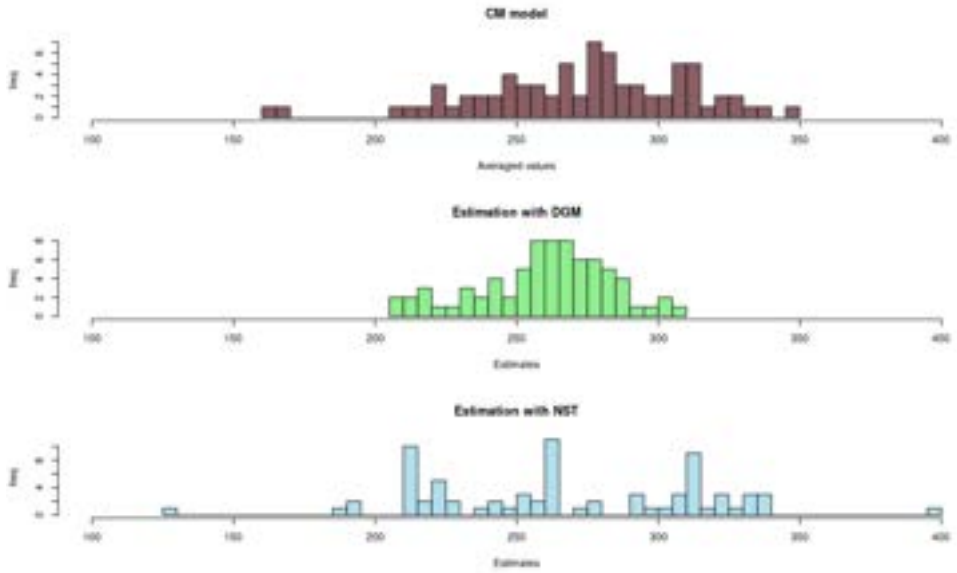


Figure 2.12: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model as in Figure 2.6. The number of sampled points is $N = 40$ and the function in (2.33) has as variance of ϵ equal to fifty ($\sigma^2 = 50$).

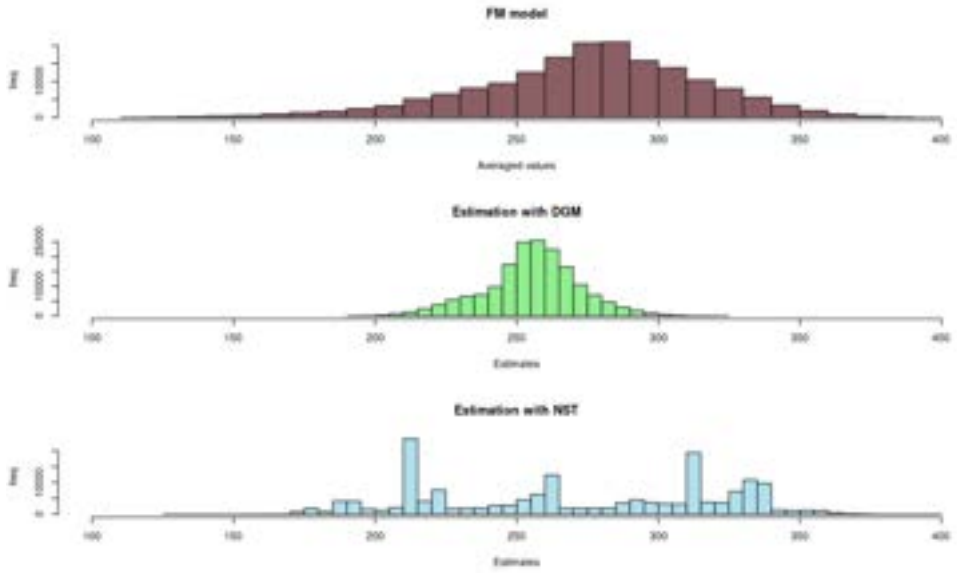
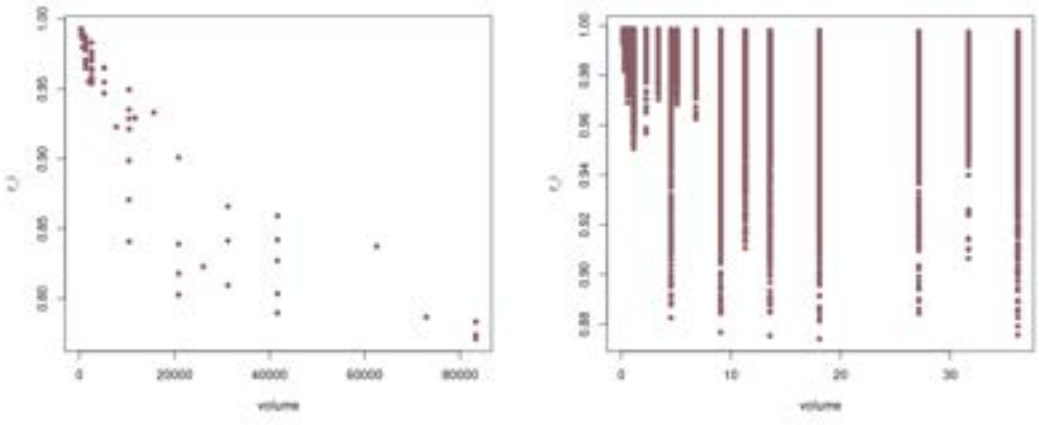


Figure 2.13: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the fine model as in Figure 2.6. The number of sampled points is $N = 40$ and the function in (2.33) has as variance of ϵ equal to fifty ($\sigma^2 = 50$).

	CM model				FM model			
	$(\mu, \sigma^2) = (0, 1)$		$(\mu, \sigma^2) = (0, 50)$		$(\mu, \sigma^2) = (0, 1)$		$(\mu, \sigma^2) = (0, 50)$	
	DGM	NST	DGM	NST	DGM	NST	DGM	NST
N=40	43962.7	70729.4	34353.2	61516.4	112576	165356	79618.5	99035.1
N=100	11316.8	26590.5	30092.1	53242.1	68210.1	91565.7	52662.4	83373.3
N=200	13122.7	24827.3	28811.8	43512.3	61355	84550.1	52155	76384.9

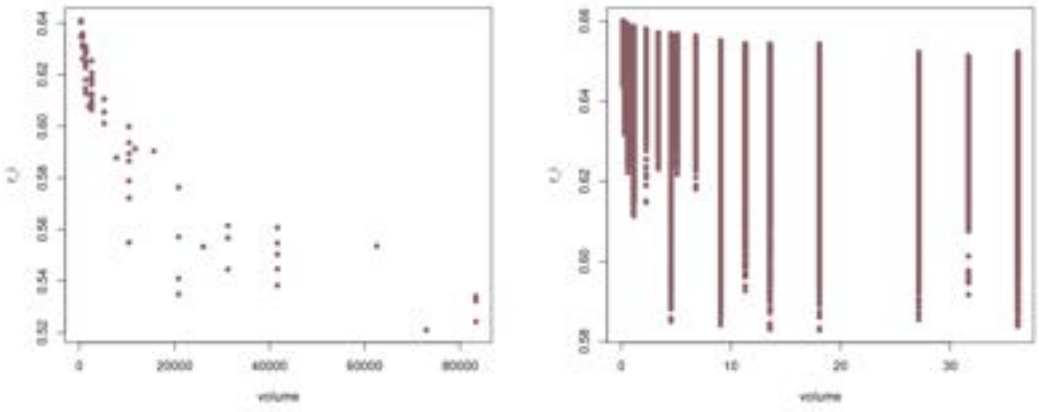
Table 2.4: MSE values of the comparison between the estimates of each model with the FM and CM averaged synthetic field respectively (Figure 2.7, 2.8) using several combinations of numbers of sampled points, type of back-transformation approach and magnitude of small-scale variability.

The values of change of support coefficients computed with DGM with respect to the volumes of tetrahedra are shown in Figure 2.14(a) and 2.14(b), for the CM and FM using $\sigma^2 = 1$ and in Figure 2.14(c) and 2.14(d) for the CM and FM using $\sigma^2 = 50$. The expected behaviour is that the smaller the volume of a tetrahedron, the closer to 1 the coefficient associated to it. This trend is fairly respected for the coarse model with both low and high value of small-scale variability (refer to the nugget values in Table 2.3). In the FM, due to the very small size of the tetrahedra of the model compared to the ranges of the variograms (61.558 and 91.792), the differences between the coefficients are not so evident and the decreasing trend is not highlighted. In some cases, a very stretched shape of the tetrahedra could give values that are not in accordance with the decreasing trend. For example, with an anisotropic spatial distribution, the tetrahedra stretched in the direction of maximum variation could have a change of support coefficient farther from 1 with respect to the same tetrahedra stretched in the perpendicular direction. In this framework the introduction of directional variogram would be crucial. In future this could be an interesting topic for further research and implementations.



(a) Coarse Model, $\sigma^2 = 1$

(b) Fine Model, $\sigma^2 = 1$



(c) Coarse Model, $\sigma^2 = 50$

(d) Fine Model, $\sigma^2 = 50$

Figure 2.14: Change of support coefficients VS volumes of tetrahedra, using $N = 40$ samples.

Chapter 3

Block-to-Block Covariance Computation

In the previous chapter, it has been shown that the computation of change of support coefficients when performing geostatistic methods on unstructured grids is crucial. This falls into the problem of computing the block-to-block covariance that should be solved with numerical integration techniques [Press et al., 1996]. Quasi-Monte Carlo methods can be used for the computation of multidimensional integrals through low-discrepancy sequences generators. Usually, low-discrepancy sequences' algorithms generate points in a unitary cube $[0, 1]^3$; however, when unstructured grids of tetrahedra are used, an extension is needed. This will allow the generation of the sequence of points directly inside the tetrahedron.

This problem has been addressed during the thesis work with a solution developed that solves the problem by mapping $3D$ points from a cube to a tetrahedron. A new technique to perform this mapping has been devised and will be discussed in this chapter.

With respect to existing solutions to solve the same problem, this approach allows to avoid the waste of generating a point in the cube and then to check if the point is inside of the tetrahedron and if not, discard it or throw it away [Zaytsev et al., 2016]. In this new implementation, all generated points are used for the numerical integration.

Moreover, a measure of discrepancy for a tetrahedron domain has been defined to test if the low-discrepancy feature is maintained after the folding. Some tests are developed to evaluate this property. Finally, a comparison among two quasi-random methods, pseudo-random generator and Centroidal Voronoi Tesselation [Nocedal and Wright, 1999], is proposed in order to verify which is the best procedure to generate points in a tetrahedron.

Section 3.1 provides an overview of the quasi-Monte Carlo method, while Section 3.2 describes how this method can be extended to deal with tetrahedral support. In Section 3.3 another approach to generate points inside a volume is presented: the Centroidal Voronoi Tessellation. Finally, several tests are performed in Section 3.4 to define the best approach to generate points inside tetrahedra and subsequently compute the block-to-block covariance.

3.1 Quasi-Monte Carlo Methods

Let v_p and v_q be two blocks with volume $|v_p|$ and $|v_q|$ respectively. The covariance between these two blocks is defined as:

$$C(v_p, v_q) = \frac{1}{|v_p||v_q|} \int_{v_p} \int_{v_q} C(x, x^{'}) dx dx^{'}$$

The computation of the block-to-block covariance is essential for determining the change of support coefficients for each element of the unstructured grid in DGM, as seen in Chapter 2.

The computation of multidimensional integrals can be effectively performed with several methods, including Monte Carlo integration techniques or their alternative variants. In numerical analysis, the quasi-Monte Carlo method is used as method for numerical integration and solving some other problems using low-discrepancy sequences (also called quasi-random sequences) [Press et al., 1996].

Traditional Monte Carlo and quasi-Monte Carlo methods are stated in a similar way. The problem is to approximate the integral of a function f as the average of the function evaluated in a set of points $x_1, \ldots, x_N$:

$$\int_D f(u) du \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

Since the domain of integration is $D \subset \mathbb{R}^d$, each x_i is a vector of d elements. The difference between quasi-Monte Carlo and Monte Carlo is how the x_i are chosen. Quasi-Monte Carlo uses a low-discrepancy sequence, whereas Monte Carlo uses a pseudo-random sequence.

The main advantage of using low-discrepancy sequences is a faster rate of convergence. The error of the approximation by the quasi-Monte Carlo method is $O\left(\frac{(\log N)^d}{N}\right)$, whereas the Monte Carlo method has a probabilistic error of $O\left(\frac{1}{\sqrt{N}}\right)$. Hence, the quasi-random sequence reaches the convergence value faster than the pseudo-random sequence [Press and Teukolsky, 1989].

The quasi-random sequence has an advantage with respect to the pure random sequence: it covers the domain of interest quickly and equally. Quasi-random sequence has an advantage also over deterministic methods. Deterministic methods give high accuracy results only when the number of points is selected a priori, whereas in using quasi-random sequence the accuracy typically improves continually as more points are added, with full reuse of the existing points. Furthermore, quasi-random point sets can have a significantly lower discrepancy for a given number of points than random sequences.

There are many methods to generate quasi-random sequences. They can be categorised using the method of constructing their basis (hyper)-parameters:

- irrational fractions: Kronecker [Beck, 1994], Richtmyer [Richtmyer, 1951], Ramshaw [Ramshaw, 1981];
- (co)prime numbers: Van Der Corput [Van Der Corput, 1935], Halton [Halton, 1960], Faure [Faure, 1986];
- irreducible Polynomials : Niederreiter [Niederreiter, 1978];
- primitive polynomials: Sobol [Sobol, 1976].

Usually, one of the most used method to generate quasi-random numbers is Sobol's algorithm [Sobol, 1976]. Sobol' sequences possess the best qualities of uniform distributiveness than any other sequence of points in the multidimensional cube $[0, 1]^d$. A significantly faster algorithm for computing points of Sobol's uniformly distributed sequence is presented by [Antonov and Saleev, 1979]. In a 3D framework, this algorithm generates points only in the unit cube. For this reason, a new approach to overcome this limitation and to extend it to generate points also in a tetrahedral support will be presented next.

Since we want to investigate also the preservation of low discrepancy property of the quasi-random sequences, another method besides that of Sobol's has been investigated. In [Roberts, 2019] quasi-random sequences algorithms to generate points in a triangle are compared: among the three different methods to triangulate and the numerous different quasi-random sequences to choose from, the parallelogram method applied to the R_2 sequence method is the unique combination that consistently produces acceptable results in terms of preserving low discrepancy without aliasing. The Recurrence R-sequence is the one that falls into the category of irrational fractions as it is a recurrence method based on irrational numbers. In particular, the irrational number that gives the lowest possible discrepancy is $1/g$, where g is the Golden Ratio ($g = \frac{\sqrt{5}+1}{2} \simeq 1.61803398875\ldots$).

Both quasi-random sequences generators, Sobol' sequence and R-sequence with Golden Ratio (GR), generate a set of values $X = \{x_1, x_2, \dots\ ;\ x_i \in \mathbb{R}^d;\ 0 < x_{ik} < 1,\ k = 1, \dots, d;\}$ with low-discrepancy over the unit interval (if $d = 1$), over the square (if $d = 2$) and over the cube (if $d = 3$). A description of both algorithms is in Appendix A.

3.1.1 Discrepancy on $[0, 1]^d$

In numerical integration the magnitude of the integration error depends also on the quality of the point-set employed for the integration. A measure of this quality will be the uniformity of the point-set. Quasi-random number generators produce a sequence of $d-$tuples that fills $d-$space more uniformly than uncorrelated random points. The discrepancy is used as a criteria of uniformity of points in the domain D. The smaller the discrepancy, the better the spacing.

The discrepancy of a set of N points $X = \{x_1, ..., x_N\}$ is defined as

$$D_N(X) = \sup_{B \in J} \left| \frac{A(B; X)}{N} - \lambda_d(B) \right|, \tag{3.1}$$

where λ_d is the $d-$dimensional Lebesgue measure (for $d = 1$, 2, or 3, it coincides with the standard measure of length, area, or volume, respectively), $A(B; X)$ is the number of points in X that fall into B, and J is the set of $d-$dimensional intervals or boxes of the form

$$\prod_{j=1}^{d} [a_j, b_j) = \{\mathbf{x} \in \mathbf{R}^d : a_j \leq x_j < b_j;\ j = 1, \dots, d\}, \tag{3.2}$$

where $0 \leq a_j < b_j \leq 1$. The star-discrepancy $D_N^*(X)$ is defined similarly, with the exception that the supremum is taken over the set J^* of rectangular boxes of the form

$$\prod_{j=1}^{d} [0, u_j) \tag{3.3}$$

where u_j is in the half-open interval $[0, 1)$. The discrepancy computation become:

$$D_N^*(X) = \sup_{B \in J^*} \left| \frac{A(B; X)}{N} - \lambda_d(B) \right| \tag{3.4}$$

When the domain to be evaluated is a square (or cube if $d = 3$), the discrepancy value is directly calculated, but if the shape of the domain is different (e.g. a tetrahedron), the calculation must be redefined. In this thesis the concept of discrepancy

is extended to tetrahedra.

3.2 Quasi-Random Sequences in a Tetrahedron

A quasi-random sequence generates points in the unit cube (in $1D$ it is an interval and in $2D$ it is a square). For this, in previously works as in [Zaytsev et al., 2016], the quasi-Monte Carlo approach requires an additional subroutine to compute the block-to-block covariance on supports different from the unit cube. There are two common approaches to handle this problem and both use the bounding boxes B_p of volume v_p and B_q of v_q:

1. Extend the integrand function $C(x, x^{'})$ to the region $B_p \times B_q$:

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

   and perform Monte Carlo integration on $B_p \times B_q$.

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

The first option does not reject the generated pseudo-random (quasi-random) points, but it increases the variance of the result due to have considered the points outside $v_p \times v_q$. The second option has an additional computation penalty for finding the points in the interior of $v_p \times v_q$. It follows that in the first case there is a waste of points and in the second a waste of time.

To avoid this waste, it is necessary to extend the methods that generate quasi-random sequence and to generate points directly on the support of interest (i.e. tetrahedron). In this way, the whole set of points is used in the estimation of the block-to-block covariance and the new strategy permits to be faster according to computational times.

Starting with a $2D$ problem, points on the triangle can be computed by mapping a pseudo-random (or quasi-random) sequence from the unit square to the triangle surface. The mapping to the triangle surface is computed according to Turk's proposals [Turk, 1990]. There are two possible methods to map these points. Given three points A, B and C that describe a triangle, a random point in that triangle

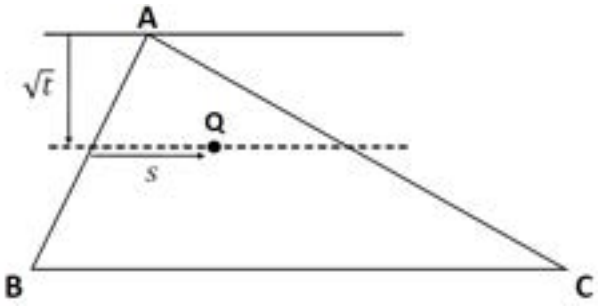


Figure 3.1: Random point Q in triangle of vertices ABC using Method-1.

is picked. When many such points are picked, the distribution of points should be uniform across the triangle.

Method-1

Let s and t be two numbers chosen from a uniform distribution of random numbers in the interval $[0, 1]$. Then, the point $Q = aA + bB + cC$ is a random point in the triangle with vertices A, B and C:

$$\begin{cases} a & = 1 - \sqrt{t} \\ b & = (1 - s)\sqrt{t} \\ c & = s\sqrt{t} \end{cases}$$

This is equivalent to having t that determines a line segment parallel to BC that joins a point on AB with a point on AC, and then picking a point on this segment based on the value of s (see Figure 3.1).

Taking the square root of t is necessary to weight all portions of the triangle equally. The values s and t are the coordinates of the point in the unit square, while the values a, b and c are the barycentric coordinates for the point in the triangle.

Method-2

Let s and t be random numbers in $[0, 1]$. A random point $Q = aA + bB + cC$ in the triangle is given by the following steps:

- if $(s + t) > 1$ then $s = (1 - s)$ and $t = (1 - t)$
- then:

$$\begin{cases} a & = 1 - s - t \\ b & = s \\ c & = t \end{cases}$$

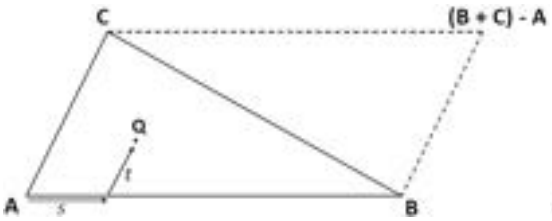


Figure 3.2: Using Method-2 when $s + t \leq 1$.

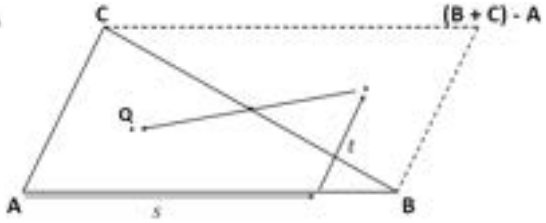


Figure 3.3: Using Method-2 when $s + t > 1$.

Without the "if" statement, the point Q will be a random point in the parallelogram with vertices A, B, C and $(B + C) - A$ (see Figures 3.2 and 3.3). A point that lands in the triangle B, C, $(B + C) - A$ is moved into the triangle A, B, C by reflecting it about the center of the parallelogram.

Method-1 can be extended to higher-dimensional shapes in a straightforward manner. Let s, t and u be three numbers chosen from a uniform distribution of random numbers in the interval $[0, 1]$. Then, the point $Q = aA + bB + cC + dD$ is a random point in the tetrahedron with vertices A, B, C and D:

$$\begin{cases} a &= (1 - \sqrt{t})\sqrt[3]{u} \\ b &= (1 - s)\sqrt{t}\sqrt[3]{u} \\ c &= s\sqrt{t}\sqrt[3]{u} \\ d &= 1 - \sqrt[3]{u} \end{cases}$$

The values s, t and u are the three coordinates of the point in the unit cube, while the values a, b, c and d are the barycentric coordinates for the point in the tetrahedron. The cube root of u is used to pick a triangle that is parallel to the base of the tetrahedron (see Figure 3.4) and s and t are used to pick a random point on that triangle as described above.

On the other hand, Method-2 has been extended by [Rocchini and Cignoni, 2000].

Depending on which technique for the generation of quasi-random sequence is selected, one of the two methods is applied to that sequence in order to obtain points directly on tetrahedron.

After the folding of points into the tetrahedron, however, this does not imply that such transformation will preserve the even spacing (i.e. low discrepancy) of the quasi-random point distributions. This obviously breaks the critical low discrepancy structure and the effect will appear biased. The discrepancy, as described in Section 3.1, can not be applied in a context different from the cubic support. So,

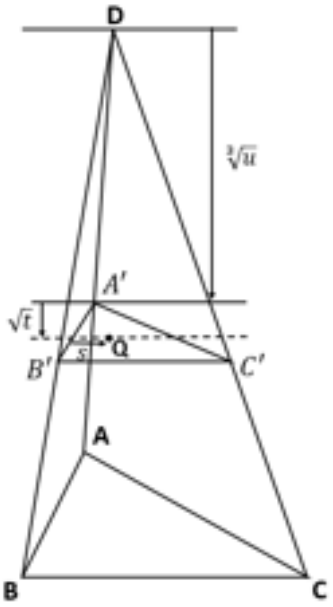


Figure 3.4: Random point Q in a tetrahedron of vertices $ABCD$ using Method-1.

for evaluating the property of optimal spacing in a tetrahedron, a new definition of discrepancy is presented below.

3.2.1 Discrepancy on Tetrahedron

To evaluate behaviour of discrepancy in 3D context with tetrahedra, a new implementation for the computation of it is needed. In order to obtain a value equivalent to one of the discrepancies in the cube, it is necessary to understand how to build the sub-volumes inside the tetrahedron, as the sub-rectangles are defined for the cube.

The first choice for defining the sub-tetrahedron is to select which vertex of the tetrahedron has to be picked (the corresponding $(0, 0, 0)$ of the cube for $D_N^*(X)$). This vertex will be called *origin vertex*. The second choice is if the point of quasi-random sequence belongs to the face opposite to the *origin vertex* (case 1) or if it is one of the vertices of the sub-tetrahedron (case 2). In the case 1, once the *origin vertex* is fixed, the point of quasi-random sequence is used to build the sub-tetrahedron in this way: the plane parallel to the face opposite to *origin vertex* that passes through the point of quasi-random sequence defines one face of sub-tetrahedron. The points where this plane intersects the edges of the original tetrahedron are the vertices of the sub-tetrahedron (see Figure 3.5). In this way, varying the four *origin vertex*, four possible scenarios are possible to obtain the discrepancy values for the distribution of points into the tetrahedron.

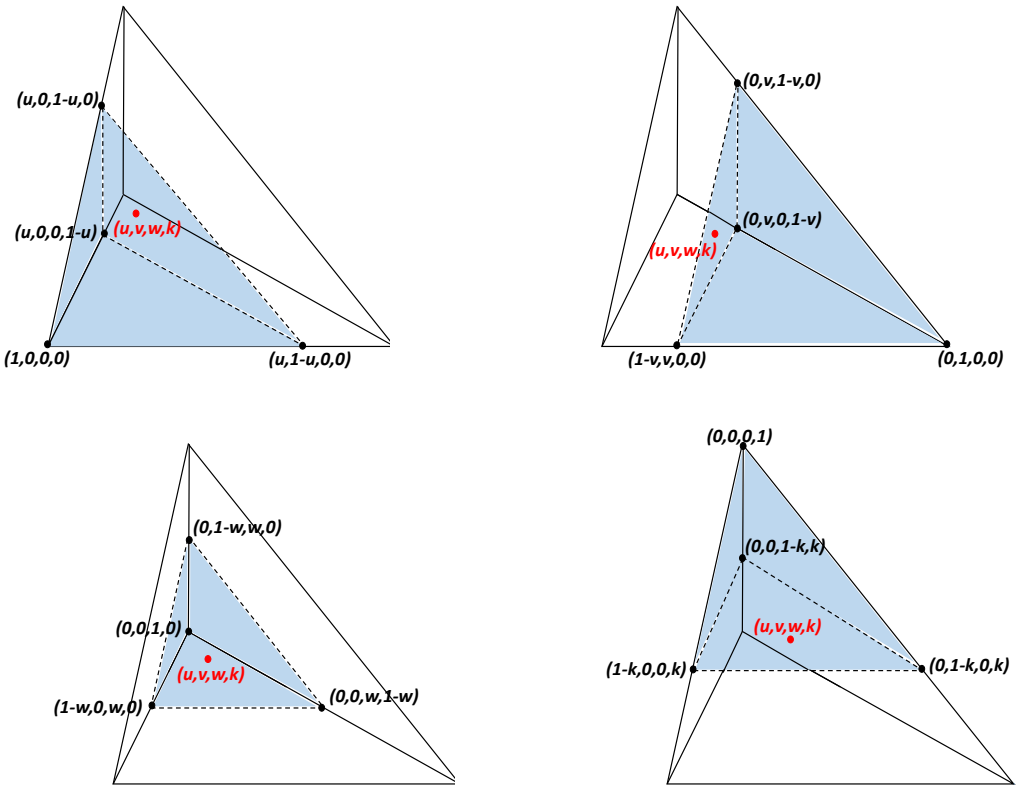


Figure 3.5: Sub-tetrahedra generated if the point of sequence (with barycentric coordinates (u, v, w, k)) belongs to the plane opposite to *origin vertex*: top left, *origin vertex* $= (1, 0, 0, 0)$; top right, *origin vertex* $= (0, 1, 0, 0)$; bottom left, *origin vertex* $= (0, 0, 1, 0)$, bottom right, *origin vertex* $= (0, 0, 0, 1)$.

Otherwise, if the point of sequence is a vertex of sub-tetrahedron, (case 2), the scenarios that can be happen will be three (see Figure 3.6):

- **Configuration A** : the vertices of the sub-tetrahedron are: *(i)* the *origin vertex*; *(ii)* the point of the sequence, *(iii)* the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment AB; *(iiii)* and the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment BC

- **Configuration B** : the vertices of the sub-tetrahedron are: *(i)* the *origin vertex*; *(ii)* the point of the sequence, *(iii)* the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment AB; *(iiii)* and the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment CD

- **Configuration C** : the vertices of the sub-tetrahedron are: *(i)* the *origin vertex*; *(ii)* the point of the sequence, *(iii)* the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment BC; *(iiii)* and the point where the plane parallel to the face opposite to *origin vertex* and passing through the point of the sequence intersects the segment CD

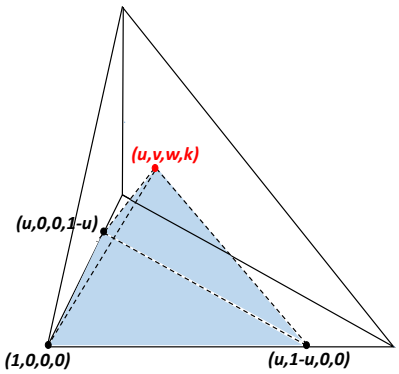
Using the second procedure to build sub-tetrahedron the possible configurations to compute the discrepancy value are twelve: for each *origin vertex* ($(1,0,0,0)$, $(0,1,0,0)$, $(0,0,1,0)$ or $(0,0,0,1)$) there are three configurations types (A, B and C).

The role of the type of strategy used for the definition of sub-tetrahedrons for the evaluation of the discrepancy in the tetrahedron is really important and could lead to different results. This will be shown in Section 3.4.

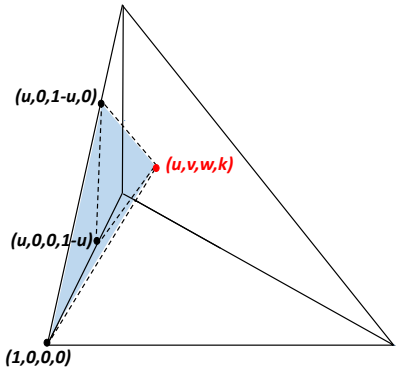
Finally, the discrepancy value of a set of N points $X = \{x_1, ..., x_N\}$ in a tetrahedron is computed as:

$$\Delta_N(X) = \sup_{subT \in \tau} \left| \frac{A(subT; X)}{N} - Vol(subT) \right| \tag{3.5}$$

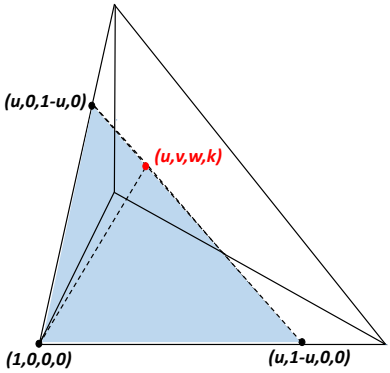
where $Vol(subT)$ is the volume of the sub-tetrahedron, $A(subT; X)$ is the number of points in X that fall into the sub-tetrahedron $subT$, and τ is the set of sub-tetrahedra that could be defined in different ways as explained above.



(a) Configuration A



(b) Configuration B



(c) Configuration C

Figure 3.6: Sub-tetrahedrons if the point of sequence (with coordinates (u, v, w, k)) is one of the vertex of the sub-tetrahedron. In this case the *origin vertex* is $(1, 0, 0, 0)$.

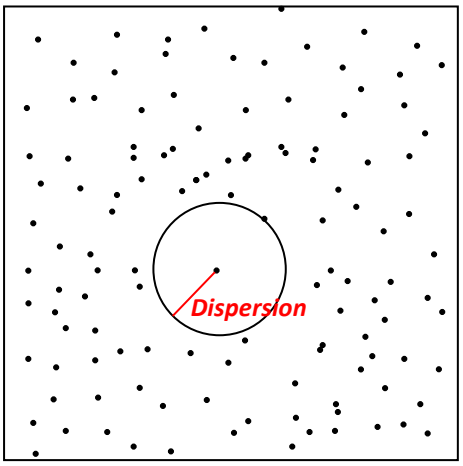


Figure 3.7: Dispersion on a sequence of random points on a unit cube.

Moreover, it is possible also to compute the values of discrepancy using the average and not the $\sup|f(.)|$ function. This new discrepancy measure is named Mean-Discrepancy. For a set of N points $X = \{x_1, ..., x_N\}$, the values of the Mean-Discrepancy for a tetrahedral support are:

$$\bar{\Delta}_N(X) = \frac{1}{N} \sum_{subT \in \tau} \left| \frac{A(subT; X)}{N} - Vol(subT) \right| \tag{3.6}$$

The Mean-Discrepancy will be used during the testing phase.

3.2.2 Dispersion Measure

The discrepancy measure is not the one and only possible criterion to characterise uniformity of a sequence of points in a subset of Euclidean space. Dispersion (or covering radius) is another estimator of spread of a sequence used in numerical optimisation [Gandar et al., 2010]. Contrary to discrepancy, dispersion is not a measure but is a criterion based on distance. Now, we consider unit cube $I_d = [0, 1]^d$ with the Euclidean distance θ.

Dispersion of a sequence $X = \{x_1, ..., x_N\}$ is defined by:

$$\Theta(x) = \sup_{y \in I_d} \min_{i=1,...,N} \theta(y, x_i)$$

Intuitively dispersion of a sequence is the radius of the biggest empty ball of I_d and Figure 3.7 shows it graphically.

Discrepancy and dispersion are not equivalent measures. Indeed, every time

a point is added to a sequence, its dispersion can only decrease, whereas its discrepancy can increase or decrease. Moreover, for an appropriate number of points, the configuration which minimises the dispersion is a regular grid which does not minimise the discrepancy.

3.3 Lloyd's Relaxation

Not only low-discrepancy sequences methods will be evaluated to provide points inside a general domain, but also other techniques, as for example Centroidal Voronoi Tesselation, could give some interesting results in the framework of generation of evenly spaced sets of points in a subsets of Euclidean space.

In geometry, a Centroidal Voronoi Tessellation (CVT) is a special type of Voronoi diagram. A Voronoi tessellation is called centroidal when the generating point of each Voronoi cell is also its centroid [Du et al., 1999].

A Voronoi diagram is a partition of the plane into j convex polytopes. Each partition contains one generator such that every point in the partition is closer to its own generator than any other generator. The constraint for the Centroidal Voronoi Tessellation is simply that each Voronoi generator must be the mass centroid for its corresponding Voronoi region.

The advantage of using this method is that the shape of the support in which to generate the points is not binding, and any support can be managed directly.

3.3.1 Lloyd's algorithm

Several algorithms can be used to generate Centroidal Voronoi Tessellations, including Lloyd's algorithm [Preparata and Shamos, 2012]. This is an algorithm finding evenly spaced sets of points in subsets of Euclidean spaces and partitions of these subsets into well-shaped and uniformly sized convex cells. It repeatedly finds the centroid of each set in the partition and then re-partitions the input according to which of these centroids is the closest.

Lloyd's algorithm starts by an initial placement of j point sites in the input domain. They may be placed at random. It then repeatedly executes the following relaxation step:

- The Voronoi diagram of the j sites is computed.
- Each cell of the Voronoi diagram is integrated, and the centroid is computed.
- Each site is then moved to the centroid of its Voronoi cell.

Since Voronoi diagram construction algorithms can be highly non-trivial, especially for inputs of dimension higher than two, the steps of calculating this diagram and finding the exact centroids of its cells may be replaced by an approximation.

Each time a relaxation step is performed, the points are left in a slightly more even distribution: closely spaced points move farther apart, and widely spaced points move closer together. In this way, it is possible to find a sequence of j points in the domain that are uniformly spaced, but in this case the low-discrepancy property is not guarantee as for quasi-random sequences.

However, the algorithm converges slowly or, due to limitations in numerical precision, may not converge. Therefore, real-world applications of Lloyd's algorithm typically stop once the distribution is "good enough". One common termination criterion is to stop when the maximum distance moved by any site in an iteration falls below a preset threshold.

3.4 Testing

In this section some tests are performed to understand which is the behaviour of the discrepancy after the mapping of the points from the cube to the tetrahedron and to select the best strategy for the generation of points inside tetrahedra.

Firstly, it will be evaluated which is the best of the two methods described above (Method 1 vs Method 2); secondly, it will be assessed the effect of the choice of the *origin vertex* and of the type of configuration to define sub-tetrahedra for the computation of the new discrepancy measure. Moreover, a comparison between algorithms to generate points inside tetrahedra will be provided. Finally, a criterion for the selection of the number of points to be generated will be designed exploiting the dispersion measure.

The tetrahedron used in these experiments is the one inscribed in the unit cube that is a platonic solid. In particular, this tetrahedron can be defined by four vertices (in barycentric coordinates): $(1,0,0,0), (0,1,0,0), (0,0,1,0)$ and $(0,0,0,1)$.

3.4.1 Comparing Method-1 and Method-2

In Section 3.2 two methods have been shown in order to obtain a sequence of points directly in the tetrahedral support. Now a graphical comparison between these is necessary.

In this test the approach where the point of the sequence belongs to the plane opposite to *origin vertex* (case 1) is used, varying the *origin vertex* among the four vertices of the tetrahedron. The Mean-Discrepancy values, $\bar{\Delta}_N(X)$, are computed

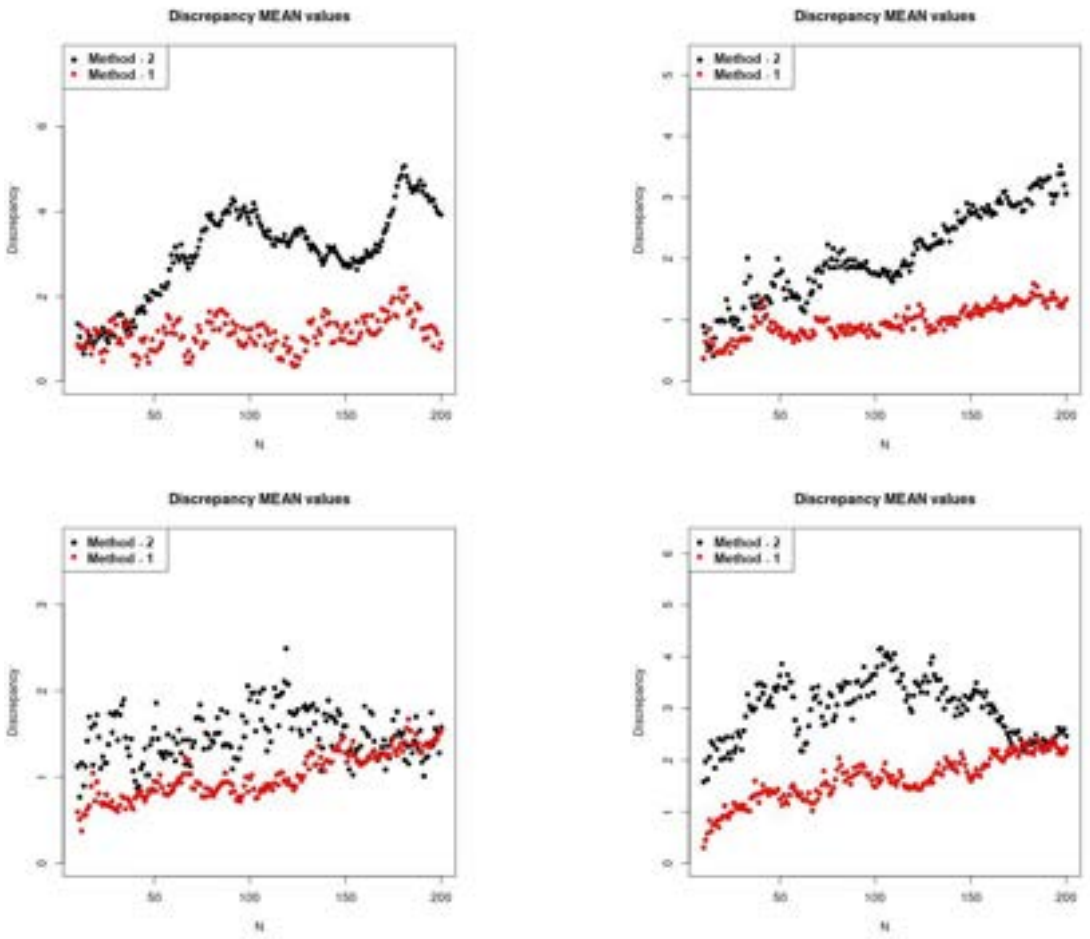


Figure 3.8: Comparison between Method-1 and Method-2 of mean-discrepancy values if the point of Sobol' sequence belongs to the plane opposite to *origin vertex*: top left, *origin vertex* $= (1, 0, 0, 0)$; top right, *origin vertex* $= (0, 1, 0, 0)$; bottom left, *origin vertex* $= (0, 0, 1, 0)$; bottom right, *origin vertex* $= (0, 0, 0, 1)$.

for a sequences of $N = 2, ..., 200$ points using Sobol (Figure 3.8) and R-sequence with Golden Ratio (Figure 3.9) methods.

Using Sobol's method to generate the sequence of points, the Mean-Discrepancy value computed with Method-1 is almost always better than that computed with Method-2; that is, the Mean-Discrepancy values of the Method-1 are smaller than those with the Method-2. It would seem that Method-1 preserves better the property of low-discrepancy of quasi-random sequence during the folding of the points in tetrahedral support.

In case the sequence of points to discretise the volume is generated with the GR method, the results are more incongruous: the choice of the *origin vertex* affects the results. The preference of one method over another is linked to the vertex chosen for the computation of the discrepancy.

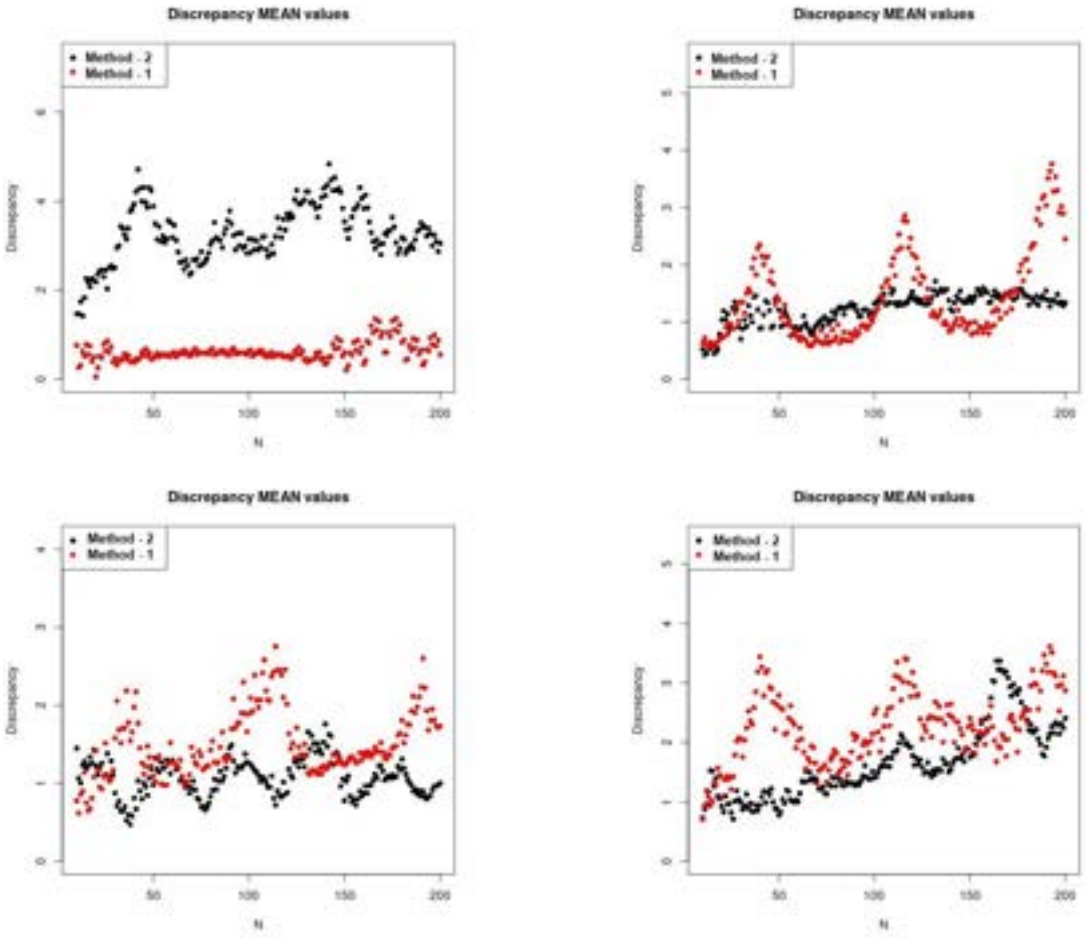


Figure 3.9: Comparison between Method-1 and Method-2 of mean-discrepancy values if the point of R-sequence with Golden Ratio (GR) belongs to the plane opposite to the *origin vertex*: top left, *origin vertex* $= (1, 0, 0, 0)$; top right, *origin vertex* $= (0, 1, 0, 0)$; bottom left, *origin vertex* $= (0, 0, 1, 0)$; bottom right, *origin vertex* $= (0, 0, 0, 1)$.

3.4.2 Analysis on the Origin Vertex

The star-discrepancy in Formula (3.4) is defined using as *origin vertex* of the cube the vertex $(0, 0, 0)$ in Cartesian coordinates. In section 3.2.1 it was shown that in order to compute the discrepancy in tetrahedron it is necessary to choose which is the *origin vertex*. For different *origin vertex*, *ov*, different values of discrepancy could be obtained. Also in the cube there exists the possibility to evaluate the discrepancy value starting from different vertex of the cube. Let $ov \in \mathbb{R}^3$ be one of the eight vertex of the cube (i.e. $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$, $(1, 1, 0)$, $(1, 0, 1)$, $(0, 1, 1)$, $(1, 1, 1)$). The new star-discrepancy $D_N^{**}(X)$ is defined over the set J^{**} of rectangular boxes of the form

$$\prod_{j=1}^{d} [ov_j, u_j) \tag{3.7}$$

where u_j is in the half-open interval $[0, 1)$.

So, for a set of N points $X = \{x_1, ..., x_N\}$, the value of Mean-Discrepancy for the cubic support are:

$$\bar{D}_N^{**}(X) = \frac{1}{N} \sum_{B \in J^{**}} \left| \frac{A(B; X)}{N} - \lambda_3(B) \right| \tag{3.8}$$

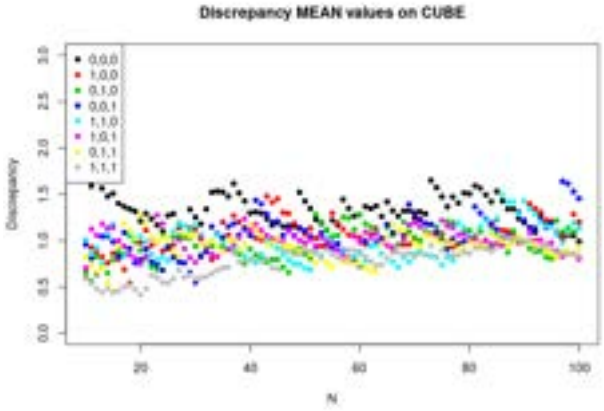
Unit Cube

To evaluate the behaviour of the discrepancy measure starting from different vertices in the unit cube, $\bar{D}_N^{**}(X)$ is computed for a sequence of $N = 2, \ldots, 100$ points generated using both Sobol's method and R-sequence with Golden Ratio irrational number (GR). They are compared with a random generator of points (Figure 3.10).

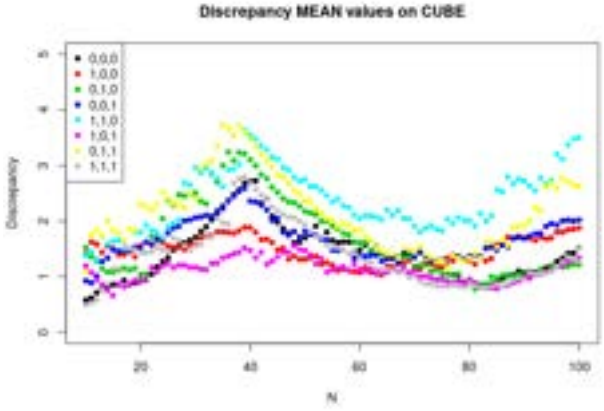
It is evident that with a sequence of points generated by Sobol's method and GR technique the discrepancy values change if the *origin vertex* is different. This is because, starting from a different vertex of the unit cube, the sub-cube that will be evaluated for the discrepancy computation will be different. Since quasi-random sequence generators select points with a particular periodicity, this behaviour is more evident then in other cases. Indeed, on the other hand, if the points generated on the cube are selected randomly the results are not so affected by the starting points of the construction of J^{**}.

"Platonic" Tetrahedron

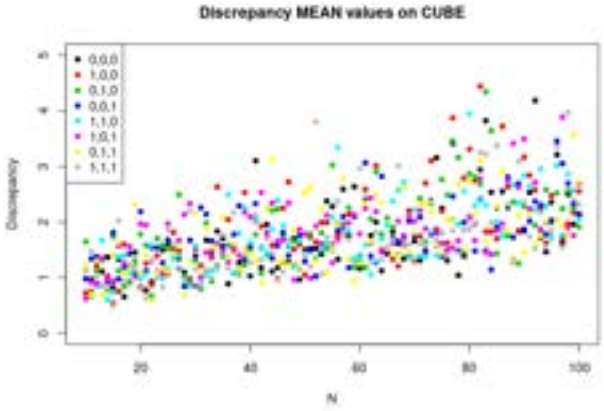
As for the cube experiment, the aim is to establish if there are differences in discrepancy values considering different *origin vertices* in tetrahedron (Figure 3.11). For this, $\bar{\Delta}_N(X)$ is computed for a sequence of $N = 2, \ldots, 200$ points generated with



(a) Sobol's method



(b) GR method



(c) Random generator

Figure 3.10: Mean-Discrepancy values for the sequences of $N = 2, \ldots, 100$ points using (a) Sobol's method, (b) R-sequence with Golden Ratio (GR) and (c) the random generator on a unit cube.

Sobol's method, R-sequence with Golden Ratio irrational number (GR), random generator and Centroidal Voronoi Tesselation (CVT), considering that the points of the sequence belong to the face opposite to *origin vertex* (case 1).

With tetrahedral support the differences in discrepancy values using several *origin vertices* are more evident then in the experiment of the cube. In Figure 3.11(a) starting from $(0, 1, 0, 0)$ or $(0, 0, 1, 0)$ the behaviour is approximately the same in terms of discrepancy values. This means that the sequence of points generated with Sobol's method has a symmetrical distribution respect to these two vertices. On the other hand, the behaviour of the Mean-Discrepancy with $(1, 0, 0, 0)$ is totally different: it has an irregular periodic trend. A periodic behaviour of discrepancy is often found for quasi-random point generators. Finally, starting from $(0, 0, 0, 1)$ discrepancy values are greater then others. This can mean an anomalous number of points near that vertex after the folding procedure.

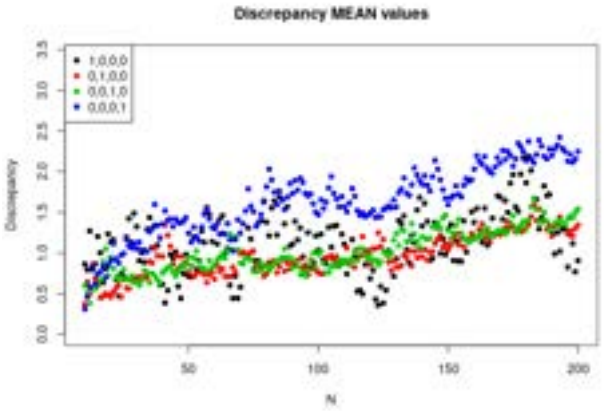
In Figure 3.11(b) results show that using as *origin vertex* $(0, 1, 0, 0)$ or $(0, 0, 1, 0)$ or $(0, 0, 0, 1)$ the values of Mean-Discrepancy of GR sequence are almost comparable: they show a periodic behaviour with the same frequency. Instead, if $(1, 0, 0, 0)$ is considered the results are very different: values are lower than others and the period is around 170 points of discretisation.

Figure 3.11(c) confirms that using a random sequence of points to discretise a volume, in terms of discrepancy, it is not matter where the *origin vertex* is fixed.

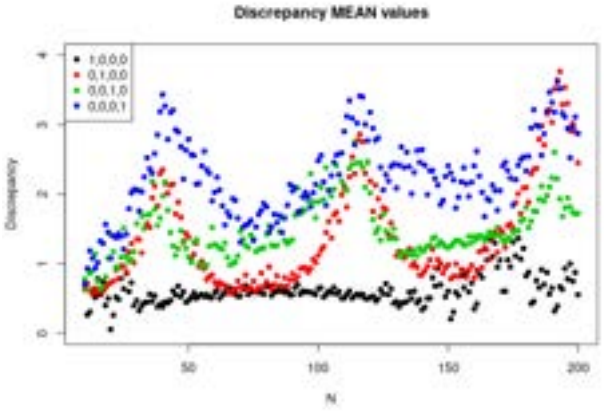
Another important comparison is between discrepancy of quasi-random sequence and Centroidal Voronoi Tesselation (Figure 3.12). CVT has discrepancy values higher than Sobol's or R-sequence with Golden Ratio methods, because this kind of discretisation does not promise property of low-discrepancy. However, for its even distribution of points, CVT has lower discrepancy values than random points generator.

In conclusion, using quasi-random sequences, the values of Mean-Discrepancy vary with the choice of *origin vertex* both in tetrahedron and unit cube. Especially in the tetrahedron domain, where there is an addition subroutine to fold points in the volume, these variations could partially affect the results. The folding operation most likely affects the regularity of the quasi-random sequences, even though the discrepancy values in the tetrahedron are similar to those in the cube.

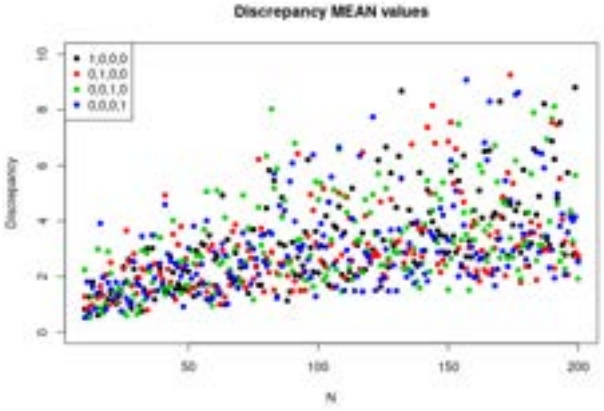
Therefore, it is important to take this aspect into account when making comparisons of discrepancy values using different generators of sequence of points.



(a) Sobol's method



(b) GR method



(c) Random generator

Figure 3.11: Mean-Discrepancy values for the sequences of $N = 2, \ldots, 200$ points using (a) Sobol's method, (b) R-sequence with Golden Ratio (GR) and (c) the random generator directly on the platonic tetrahedron.

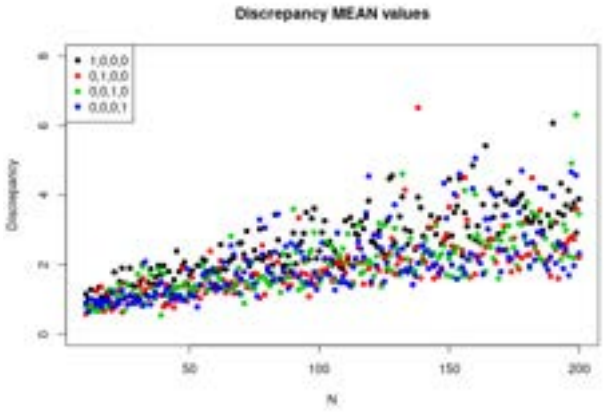


Figure 3.12: Mean-Discrepancy values for the sequences of $N = 2, \ldots, 200$ points using CVT on the platonic tetrahedron starting from different *origin vertex*.

3.4.3 Point as a Vertex

In this section, the approach of considering the point of a quasi-random sequence as a vertex of the sub-tetrahedron is tested (case 2). As seen in Section 3.1, this choice leads to three possible configurations (A, B, C) for each *origin vertex*: we want to evaluate if results are conditioned by which configuration is selected.

In Figure 3.13 and 3.14 the Mean-Discrepancy values, $\bar{\Delta}_N(X)$, are computed for a sequence of $N = 2, ..., 200$ points using Sobol's and R-sequence with Golden Ratio methods respectively. In each test the *origin vertex* varies between the four possible vertices of the tetrahedron. Using Sobol's method for the generation of the points, except if *origin vertex* is $(1, 0, 0, 0)$ where the configuration A has lower Mean-Discrepancy values than the other two configurations, the configuration does not affect significantly the discrepancy values. Instead, using R-sequence with Golden Ratio method, results are more unstable.

3.4.4 Dispersion Test

In the framework of unstructured grids they are composed by cells with several sizes. Since each volume can also be very different, a variable number of points to discretise these cells could be convenient. For a "large" cell the number of points to discretise it will be certainly bigger than for a "small" one.

However, also varying the size of tetrahedra, the Mean-Discrepancy values remain the same. For example, if $T7$ tetrahedron from Table 3.4.4 is considered, the corresponding graph of Mean-Discrepancy for sequence of $N = 2, \ldots, 200$ points with Sobol's method, is the same of Figure 3.8 where the Mean-Discrepancy was computed on the platonic tetrahedron.

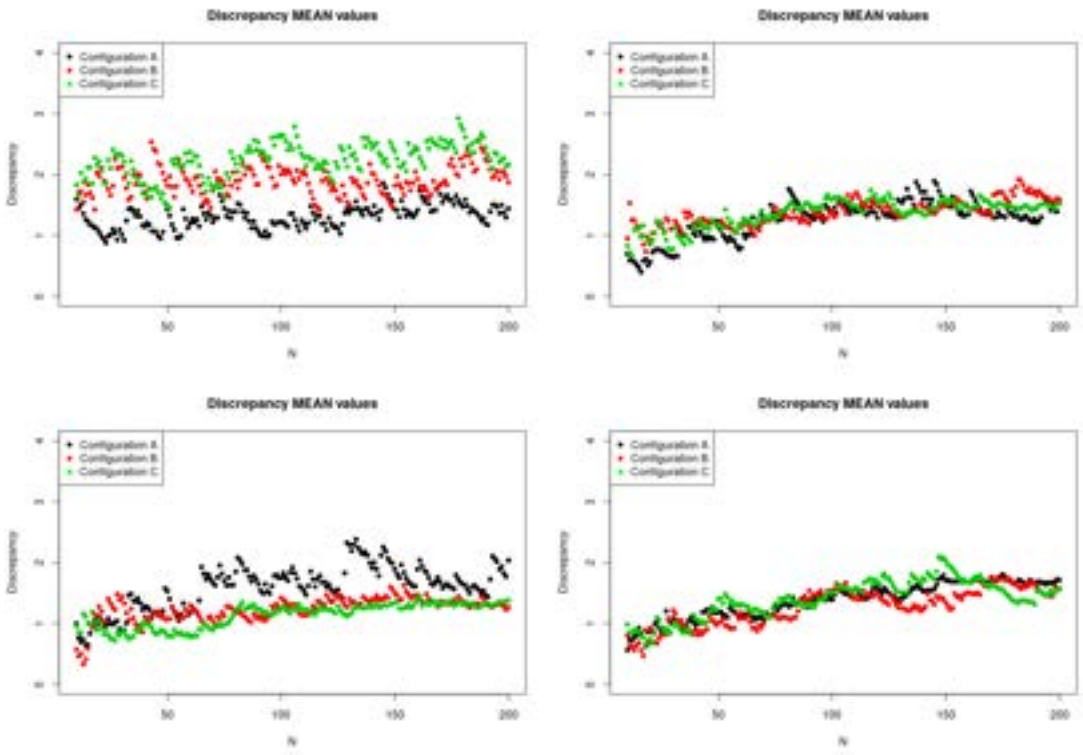


Figure 3.13: Mean-Discrepancy values when the point of Sobol' sequence is one vertex of the sub-tetrahedron: top left, *origin vertex* $= (1, 0, 0, 0)$; top right, *origin vertex* $= (0, 1, 0, 0)$; bottom left, *origin vertex* $= (0, 0, 1, 0)$; bottom right, *origin vertex* $= (0, 0, 0, 1)$.

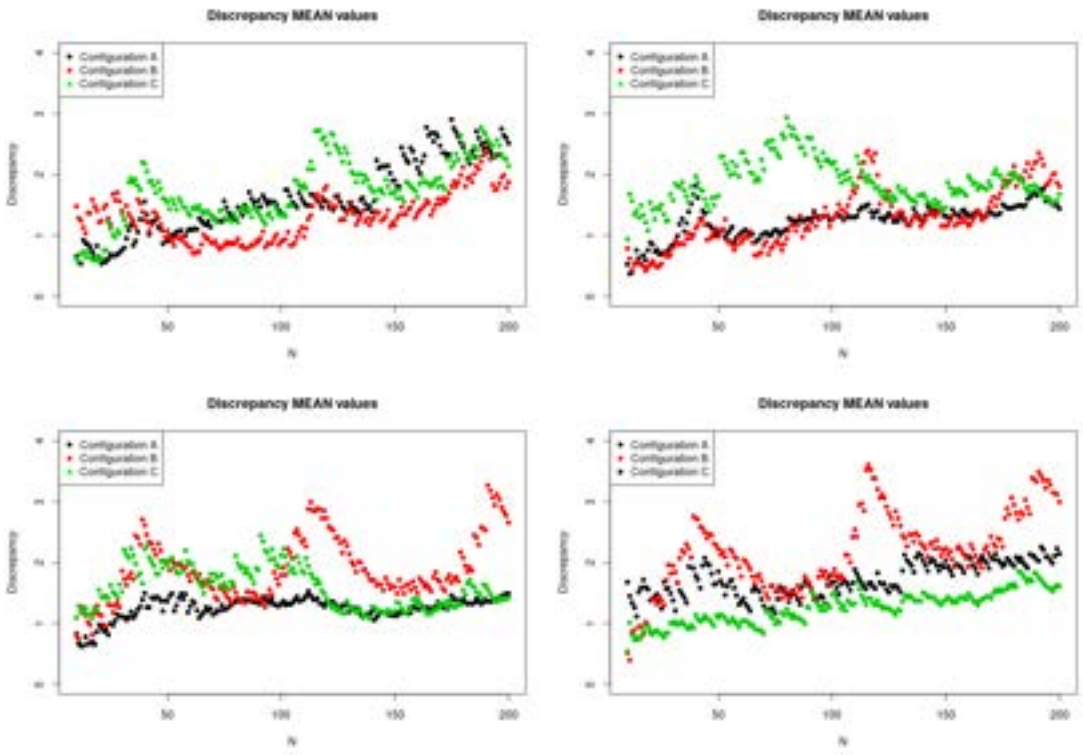


Figure 3.14: Mean-Discrepancy values when the point of R-sequence with Golden Ratio (GR) is one vertex of the sub-tetrahedron: top left, *origin vertex* $= (1, 0, 0, 0)$; top right, *origin vertex* $= (0, 1, 0, 0)$; bottom left, *origin vertex* $= (0, 0, 1, 0)$; bottom right, *origin vertex* $= (0, 0, 0, 1)$.

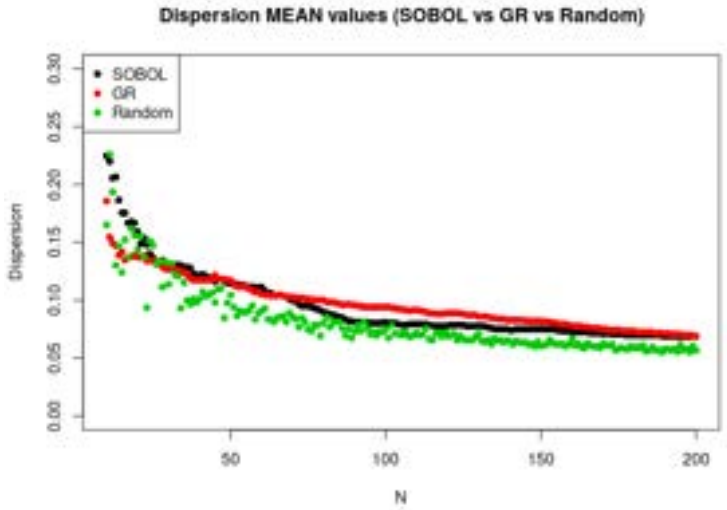


Figure 3.15: Comparison of Mean-Dispersion values among the three methods: Sobol' sequence (black), R-sequence with Golden Ratio (GR) (red), Pseudo-Random (green).

For this reason the discrepancy measure cannot help in defining a procedure for selecting the number of points to be generated in each cell of the unstructured grid. The shift from discrepancy to dispersion evaluation is also supported by the necessity to have a criterion to select the number of points to discretise the domain of interest based on its volume.

In this test, as for discrepancy, the Mean-Dispersion value is defined:

$$\bar{\Theta}(x) = \frac{1}{N} \sum_{y \in I_d} \min_{i=1,\dots,N} \theta(y, x_i)$$

It must be considered that using this definition of Mean-Dispersion, the property of decreasing monotony of the dispersion is lost. However, except for some points, the decreasing trend is kept.

Figure 3.15 shows behaviour of the Mean-Dispersion of sequences of $N = 2, \dots, 200$ points generated with Sobol's method, R-sequence with Golden Ratio and pseudo-random generator directly on the platonic tetrahedron using Method-1. The behaviour of the Mean-Dispersion is the same for all of them.

In order to test the behavior of the dispersion values varying the size of the tetrahedron, several tetrahedra are defined with different volumes. Table 3.4.4 summarises their vertices and volumes. For each tetrahedron Mean-Dispersion values, $\bar{\Theta}(x)$, are computed for sequence of $N = 2, \dots, 200$ points and results are shown in Figure 3.16.

In order to select the number of points to generate in a specific volume, a threshold of the Mean-Dispersion must be fixed. It will be the averaged distance that is

Name	Vertex #1	Vertex #2	Vertex #3	Vertex #4	Volume
$T1$	$(0,0,0)$	$(1,0,0)$	$(0,1,0)$	$(0,0,1)$	0.167
$T2$	$(5,0,1)$	$(4,3,2)$	$(1,5,0)$	$(1,0,1)$	5.333
$T3$	$(0,0,0)$	$(2,0,0)$	$(0,2,0)$	$(0,0,2)$	1.333
$T4$	$(1,3,5)$	$(3,3,2)$	$(1,3,1)$	$(2,4,0)$	1.333
$T5$	$(5,1,1)$	$(4,3,2)$	$(1,5,0)$	$(1,8,1)$	1.833
$T6$	$(5,1,1)$	$(4,3,2)$	$(1,5,9)$	$(1,8,1)$	3.333
$T7$	$(0,1,1)$	$(4,3,2)$	$(0,5,9)$	$(1,4,1)$	14.000
$T8$	$(5,1,5)$	$(4,3,2)$	$(2,5,9)$	$(2,8,1)$	3.833
$T9$	$(8,1,1)$	$(4,3,2)$	$(2,3,7)$	$(1,3,1)$	5.667
$T10$	$(1,1,1)$	$(3,0,2)$	$(0,3,0)$	$(0,3,1)$	0.500

Table 3.1: Definition of 10 tetrahedrons.

required among points in the domain. In Figure 3.16 three possible thresholds are shown with dashed horizontal lines: 0.20, 0.30 and 0.40. The number of points for the discretisation is selected accordingly to one of these threshold: as soon as the Mean-Dispersion curve reaches the selected threshold, the corresponding number of points is chosen. For example, considering the tetrahedron $T5$, the number of points to select in order to obtain an average distance among points approximately of 0.40 is about 20 (if the threshold is 0.30 or 0.20, N will be 25 or 30, respectively). If a tetrahedron with a larger volume is considered, e.g. $T7$, considering the same threshold, the selected number of points is greater than that chosen for $T5$ (about 50 points).

3.4.5 Conclusion of the Experiment

The procedure of folding with both methods described in this thesis does not guarantee the low-discrepancy property typical of the quasi-random sequences. However, we have seen that the new positioning of points in tetrahedra maintains good discrepancy values compared to pseudo-random arrangement, especially using Method-1 with Sobol' sequences.

The results of the test might suggest that Sobol' algorthm is the most robust generator of sequences, maintaining low values of discrepancy after the folding of points without high variations using different *origin vertices* or several configurations.

Finally, identifying the number of points to be generated in a sequence for the estimation of block-to-block variance, when the size of the cells is different (unstructured grid), is a practical aspect that requires a selection criterion. Discrepancy values do not fit these purposes, while dispersion values are more interesting. The definition of a threshold of the Mean-Dispersion that in each tetrahedron must be

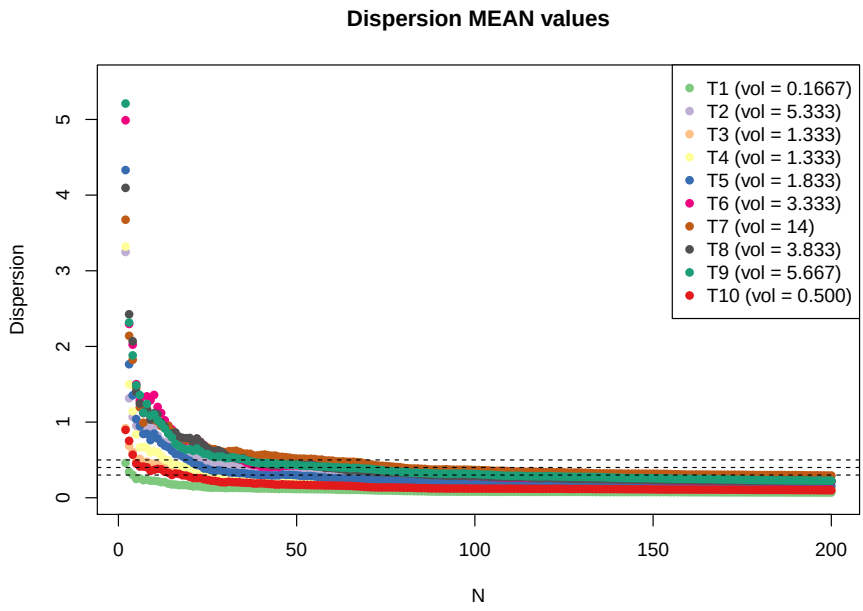


Figure 3.16: Comparison of Mean-Dispersion values computed on ten different tetrahedra with different shapes and sizes (see Table 3.4.4) with threshold 0.20, 0.30, 0.40 (dashed horizontal lines).

reached allows to identify the specific number of points of the sequence that meets this threshold. The larger the cell volume, the greater the number of points to achieve the required Mean-Dispersion target value.

Chapter 4

Adaptive Sampling Strategy

This chapter focuses on the adaptive sampling procedure optimised with respect to the uncertainty associated with the estimates of a random field $\{Z(x), x \in R^d\}$, which is a core contribution of the thesis [Berretta et al., 2018a].

In the previous chapters, the building blocks needed to set-up the new sampling strategy have been discussed and in this chapter their integration into an effective surveying strategy are presented, both in the case of regular and irregular discretisations of the geometric, or physical, domain.

This new sampling strategy allows to highly improve traditional environmental sampling: thanks to new sensor technologies for in-situ analysis, the adaptive sampling method can speed up the surveying and, at the same time, reduce the costs and waste of resources exploiting the use of real-time measurements of the variables to be analysed. The impact on environmental survey campaigns will be fully discussed in Chapter 5, while here the main steps of the procedure are presented.

Considering the potential of novel sensors, the new strategy allows to collapse into a single step the three phases of the standard strategy for environmental sampling: *(i)* the sampling design where the locations of the samples are selected a priori; *(ii)* the sampling phase where the measurements of the environmental variable are collected in the set of locations selected during the phase *(i)*; *(iii)* finally, the data analysis in which the estimation of the environmental variable distribution on the domain of interest is computed.

An important feature of the approach developed is that the reasoned arrangement of the positions builds not only a reliable estimate of the field, but also a well-documented map of its uncertainty. This is an important innovative aspect that has impact also at the application level as it allows to visualise directly the uncertainty of the data analysis process.

The chapter is organized as follows. Section 4.1 introduces the main characteris-

tics of the sampling strategies in a general context and how these can be optimised to improve the analysis. This section also introduces some aspects of traditional sampling strategies in the framework of environmental monitoring and how the new technologies can help to evolve in these applications. In Section 4.2 the adaptive procedure is shown in details, with all its phases, considering a regular discretisation of the physical domain (regular blocks). Section 4.3 discusses the necessary modification when an unstructured grid is used to represent the survey domain: here, change of support models will be integrated in the pipeline to support the use of very generic discretisation models of the physical domain.

In Section 4.4, the algorithms developed to implement the adaptive sampling are outlined. Finally, in Section 4.5 a comparison between different sampling strategies is provided in order to verify the improvement given by using an adaptive approach.

4.1 Sampling Strategies

Sampling is the technique that allows to describe a phenomenon starting from the knowledge of only a portion of it (sample). How the samples are selected plays a crucial role to ensure a high quality in the results. There are several approaches to properly select samples.

In a general perspective, when a statistical research about a group of people is conducted, it's rarely possible to collect data from every member in that group [Levy and Lemeshow, 2013]. For this reason, a sample is needed. The sample is the specific set of individuals from which to collect data, while the population is the whole group about which one wants to draw conclusions. Sampling is the technique of selecting individual to make statistical inferences from them and estimate characteristics of the whole population.

To draw valid conclusions from results, one has to carefully decide how to select a sample that is representative of the group as a whole. The selection of samples plays a crucial role to obtain relevant results from the analysis. There are mainly two types of sampling methods:

- *probability sampling* for which every unit in a finite population has a positive probability of selection, not necessarily equal to that of other units. It includes some form of random selection in choosing the elements: 1) simple random, 2) stratified random, 3) cluster, and 4) systematic [Kothari, 2004];

- *non-probability sampling* involves non-random selection based on convenience or other criteria. Common types of non-probability sampling methods in-

clude quota sampling, purposive sampling, self-selection sampling, snowball sampling and others [Etikan and Bala, 2017].

The selection of the sampling method depends on several factors, e.g. the population of interest and the type of statistical analysis the researcher is applying. Pros and cons of these sampling methods are described in several works as for instance in [Taherdoost, 2016] and [Berndt, 2020].

Even if the conventional sampling designs mentioned above could be an effective solution for many applications, nevertheless, by incorporating further information the selection of items can be guided by a more reasoned criterion. One alternative to conventional designs is the set of adaptive sampling methods. With adaptive sampling, the method of selecting observations at a given step depends upon the observations collected until the previous steps. In this case, items are selected iteratively during the procedure exploiting the information collected on previous samples and trying to optimise the reliability of the estimate of the phenomenon.

A further and important step in the sampling strategy is to decide an appropriate sample size. There are no strict rules for selecting a sample size. The decision can be based on the objectives of the research, time available, budget, and the necessary degree of precision. However, knowing some information a priori is not always possible and for this the choice of the sample size becomes challenging [Singh and Masuku, 2014]. In the following sections it will be discussed the sample size topic and how this will be determined defining a stop criterion of the adaptive procedure.

4.1.1 Environmental Sampling Strategies

The environmental sampling is the way the environment is interrogated to get measures of some parameters in a limited set of locations. In the environmental sampling, the population is represented by the infinitely places at which the parameters might be recorded, while the sample is represented by the finite positions that can be measured.

Both [Zhang, 2007] and [Keith, 2017] provide a detailed analysis of environmental sampling strategies considering several frameworks. However, in order to select the most appropriate sampling strategy it is important to distinguish between environmental variables that can be measured *in-situ* or by analysis of grab samples. If it is not possible to analyze the parameter of interest directly on site, some laboratory analyses to obtain the final measurement of a sample are needed. In these cases, traditional sampling strategies are divided in three distinct phases (Figure

Figure 4.1: Traditional environmental sampling phases when the environmental parameter can not be analyzed *in-situ*.

4.1). First of all, in the survey design phase and before the start of the survey, the positions in which to carry out the measurements are selected. Typically, sample locations are placed either randomly or on a regular grid or along directions that are selected with respect to experts' a priori knowledge about the system. Then, in the second phase, the sampling is carried out, that is, the measurements are acquired at the selected locations either by human operators or by fixed sampling stations. The measurements, usually, occur without real-time feedback on the samples that have been collected. Finally, during the third phase, the collected samples are analysed in the laboratory. During this last phase results are evaluated and experts decide if they are satisfied with them. In some cases, if the results have not high quality, further samples will be selected on sub-areas of the domain in order to add data where some problem has been encountered and a new campaign is reprogrammed.

Clearly, this procedure could be very expensive in time and cost. Since the reliability of the results can be evaluated only after the end of the survey, further campaigns to improve the estimations must be reprogrammed each time and the associated costs could significantly increase. For this reason, new approaches have been investigated to limit this waste of resources.

Nowadays, new dynamic positioning systems and lighter and cheaper sensors are available, as in the near future there will be even more, yielding to an explosion of georeferenced and highly accurate data for many different purposes. [Yamahara et al., 2019] and [Harvey et al., 2012] are two examples that exploit these cutting-edge technologies.

Having a real-time punctual measurement of an environmental variable opens the door to new on-the-fly sampling decisions. On the other hand, this requires innovative computational solutions to make data analysis precise and fast. The new proposed approach leverages on the use of these real-time sensors to investigate physical/chemical parameters rather than on laboratory analysis.

4.2 Adaptive Sampling Optimised by Spatial Uncertainty

The adaptive procedures have received more attention recently due to the availability of computing resources that have made their runtime variants increasingly feasible. The runtime adaptivity refers to a system's ability to adapt to runtime changes in its execution environment.

[Fuhg et al., 2021] provides a comparison of several adaptive sampling techniques at the state of the art in order to build proficient kriging models with as few samples as possible. These techniques aim to find pertinent points in an iterative manner based on information extracted from the current model. Therefore, the concept of adaptivity can be applied also to sampling strategies in order to select the sample units sequentially. In this way the new location is selected during the execution of the sampling, exploiting knowledge deduced from the data collected up to that moment. This allows to adapt the surveying process according to any insight that might emerge during the sampling procedure.

The focus of the thesis work is on the definition of a reasoned approach for sampling, where the arrangement of all the sampling locations is not known a priori. The idea is to set-up an iterative process where the next sample location is determined adaptively, on the base of the environmental scenario that is delineated more and more accurately at each iteration. Key to the development of an effective approach is the definition of optimisation criteria for the selection of new samples: the next sampling positions will be selected in order to minimise or maximise a certain amount of interest. In the thesis research program, we have worked out optimisation criteria based primarily on a measure of uncertainty of the estimates of the environmental variable, but also other aspects will be discussed.

4.2.1 Spatial Uncertainty

Uncertainty is a key characteristics to fully understand the results of an analysis and the quality of the reconstruction of the variable distribution ([Rocchini et al., 2011], [Caers, 2011]). The consideration and quantification of the uncertainty can be relevant in many practical applications and can be part of the data analysis chain to support decision making. The visualisation of the uncertainty itself could be an important tool for the decision-making process and to provide an effective and accurate communication of the results. The goal is to understand uncertainty, deal with it, and use this information to improve the scientific process.

In our strategy of adaptive sampling, the uncertainty becomes part of the opti-

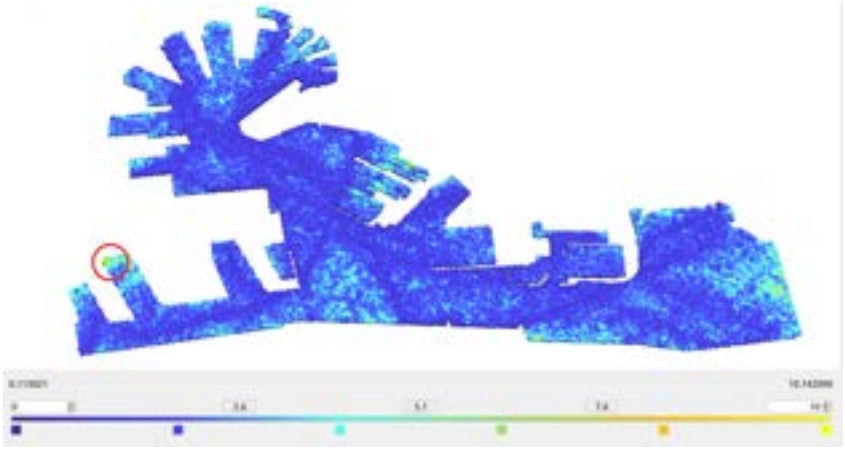


Figure 4.2: Uncertainty map.

misation criterion for the selection of new samples in the environmental sampling. The aim is to reduce the uncertainty associated to the estimates by adding points where its value is highest.

The kind of uncertainty we refer to is related to the spatial component of the arrangement of the samples on the domain, so it is called spatial uncertainty. The arrangement of positions in the volume characterises the spatial uncertainty, and depending on how these points are chosen the uncertainty will be affected.

A measure of the spatial uncertainty can be obtained from Sequential Gaussian Simulations as shown in Section 1.4.2. By simulation techniques, it is possible to generate some of "virtual realities" and produce pictures that are due to the fluctuations of the phenomenon. The spatial uncertainty can be represented by generating several of these digital models. Considering a survey domain discretised in M blocks and a set of puntual samples $\{u_1, \ldots, u_N\}$, the procedure provides an estimated value of the environmental variable and also a value of uncertainty for each cell of that domain, $\{U(x_1), \ldots, U(x_M)\}$. The criterion adopted is based on the idea that the higher the uncertainty of an estimate, the greater the error that can be made in associating that value to that position. Therefore, using the uncertainty measures, it is possible to identify the next point to be sampled, x_w, as the center of the cell with the highest uncertainty value.

$$x_w = \underset{j=1,\ldots,M}{\arg\max}\, U(x_j) \tag{4.1}$$

In this way the strategy adds samples where the estimates are more unstable in order to improve the estimation in those areas.

In Figure 4.2 an uncertainty map generated from a simulation process is represented to provide an example of the selection of the next point with the aim of

minimising the spatial uncertainty. The new location is selected at the bottom left of the survey domain (inside the red circle), corresponding to the highest value of uncertainty.

In this new approach, uncertainty totally changes its role in data analysis: very often in environmental monitoring, the uncertainty has played a "passive" role in which it was used a posteriori for the evaluation of the reliability of the results. Now, its role becomes more "active" by intervening step by step in the selection process of the sampled locations in order to minimise itself.

4.2.2 Alternative Optimisation Criteria

The uncertainty is not the unique ingredient that can be used to define the optimisation criterion of the sampling. Considering the surveying process in the wild, when the *in-situ* measures have to be collected in the field, one might want to consider also other aspects to optimise the efficacy of the surveying. For instance, the distance among consecutive sampling locations could represent an important factor to consider. If the distance is large, indeed, moving from one location to another could involve extra time and costs, and result at the end in a less effective overall survey.

Another important aspect is related to the use of the maximum uncertainty itself to identify the next sampling location. Since the uncertainty is computed as the variance of several "virtual realities" generated by SGS, often it might exhibit outliers. As mentioned in Chapter 1, the order of visiting points in the survey domain is selected randomly and this can lead to abnormal results. If uncertainty exhibits outliers, relying on the maximum value of uncertainty only could be cause of a slower convergence to the expected result.

In synthesis, criteria for the adaptive sampling could consider also:

- the displacement to reach the new points selected by the adaptive strategy step by step, avoiding that this becomes too large;

- the possible presence of outliers in the computation of the maximum value of the uncertainty, to reduce the risk they can produce an high value in the variance of simulations.

To manage the first issue, the distance to reach the new point is considered in the process of selecting the next waypoint: among the K (a parameter set by the user) points with the highest uncertainty, the one closest to the current position is selected as the new waypoint. The greater the parameter K, the shorter the

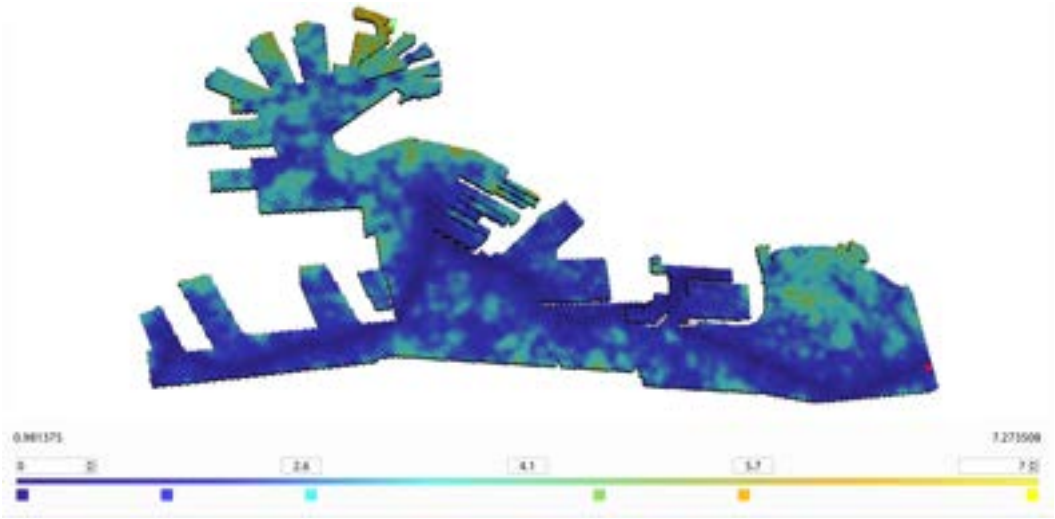


Figure 4.3: Geometric model represented with a tetrahedral mesh of a portion of Genoa harbor.

distance traveled step by step from the sampling. When $K = 1$ is the case of using only uncertainty as optimisation criterion.

In the second case, to avoid to consider outliers for selecting the next waypoint, it is possible to include the moving average (MA) method [Hyndman, 2011]. The MA method computes a series of averages of different subsets of the full data set. First of all, a parameter to set a search distance of neighboring points, τ, must be specified to define the subset of data. This parameter is used to define a sphere of center c_m, that is the centroid of the m-th cell of the geometric model ($m = 1, \ldots, M$), and radius τ. For the m-th element of the survey domain, the cells that fall within the sphere of parameters c_m and τ are identified and the average of their associated uncertainty values is computed. This averaged uncertainty is assigned to c_m. A moving average creates a smoothing effect and reduces noise of the uncertainty map. The results is a smooth representation of the spatial uncertainty. The next point to be sampled is selected according to uncertainty values computed with the MA: the maximum value of the averaged uncertainty identifies the cell of the model where address the sampling. In Figure 4.3 an example of the smoothed uncertainty map is shown.

4.2.3 Summary of the New Sampling Method

Assuming that one wants to know and represent a SRF $\{Z(u), u \in D \subset \mathbb{R}^d\}$ on a specific domain D of interest. For this, a set of samples,$\{u_1, u_2, \ldots, u_N\}$, must be collected to provide a map of the distribution of this variable. The proposed adaptive strategy is used to select these samples.

- As preliminary step, even before the start of the sampling, the geometric model for representing the survey domain must be defined. The domain D is discretised in M blocks that will be used for the modelling and visualisation of the environmental variable distribution. The coordinates of the centers of these blocks, $\{c_m, m = 1, \ldots, M\}$ will be used during the estimation phase for computing a value for each one of these centers.

- The new sampling method starts with an initial set of N_{init} random locations, $\{u_1, \ldots, u_{N_{init}}\}$, where the values of the environmental variable are collected $\{z(u_1), \ldots, z(u_{N_{init}})\}$. This represents a first set of data from which to start the iterative procedure.

- After that, this transformed data are used to obtain a preliminary experimental variogram estimation (see Section 1.2.1); then, the experimental variogram is interpolated with a function defining the type of the spatial correlation (i.e. *Spherical, Gaussian and Exponential*) and estimating *sill, range and nugget* values (see Section 1.2.2).

- Then, Sequential Gaussian Simulations are performed using the N_{init} sampled values as input in order to compute an estimation for each center of the blocks of the discretised domain D (Section 1.4.2). The estimation is obtained exploiting the information derived from the fitted variogram and the result is a vector of M elements: $\{\hat{z}(c_1), \ldots, \hat{z}(c_M)\}$. Then, the estimated values will be assigned to the entire volume of the block, generating two maps: an estimated map of the environmental variable distribution and an uncertainty map. Of course, assigning a puntual value to the entire block is an approximation and change of support models can be investigated (Chapter 2).

- At this step, an evaluation of the results is carried out to decide if it is necessary to continue with the sampling by improving the reliability of the obtained estimation. So, if the generated maps were satisfactory according to the opinion of an expert or on the basis of a criterion defined a priori, then the procedure can be concluded. Otherwise, by exploiting the uncertainty map, the new waypoint, u_w, to reach is identified. This is done selecting the position where the uncertainty is the greatest or using one of the other optimisation criterions described in Section 4.2.2.

- As next step, the sampling moves to collect the value of the spatial variable in this new location, $z(u_w)$, and adds the new sample to the set of the available

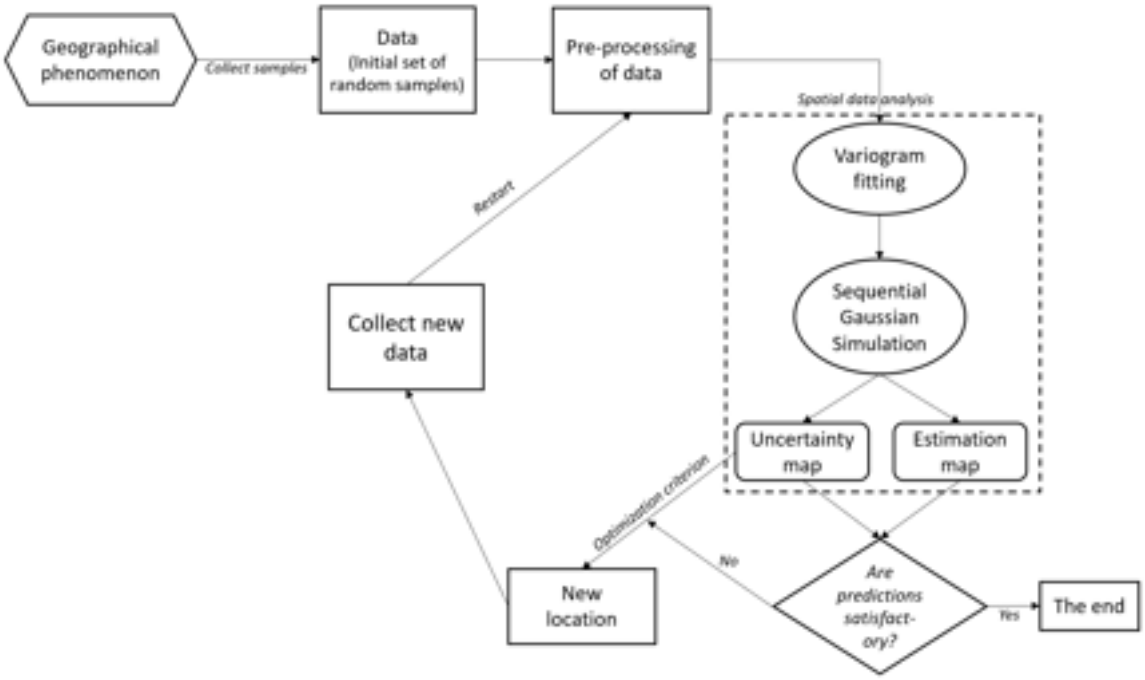


Figure 4.4: Diagram of Adaptive Sampling guided by Uncertainty Minimization.

data. So, at this step the set of sampled data is enlarged of one element: $\{z(u_1), \ldots, z(u_{N_{init}}), z(u_w)\}$

- The procedure restarts from the pre-processing phase where also the new collected data are considered and all steps are repeated.

The method is iterated until a required reliability is achieved. Figure 4.4 shows a summary diagram of this adaptive sampling process.

4.3 Adaptive Sampling Optimised by Spatial Uncertainty on Unstructured Grids

Sometimes, in environmental survey the domain is considered bi-dimensional, specifying locations using x,y coordinates and discarding elevation (e.g., for terrains but also in water, where a reference depth equal for all samples is assumed by default). While this simplification is acceptable for some kind of measures, in many fields the 3D nature of the phenomenon cannot be disregarded in order to provide a better representation of a phenomenon itself.

In spatial analysis, the use of finite volumes to represent the survey area is adopted to discretise the domain over which environmental variable distribution should be modelled first and visualised afterwards. Very often in environmental

analysis the survey domain is discretised using structured grids. Structured grids are of several varieties, depending on the shape of their elements. One of the simplest grid is generated by subdividing a rectangular box, containing the area of interest, into a set of rectangular elements whose faces are parallel to the faces of the box. Grids composed of regular cubic elements have the simplest structure, and are widely used due to their simplicity, but their limitation is the poor representation they provide of the real geometry of the physical domain: the boundaries of the physical domain, usually approximated by blocking out entire elements, would be represented by stepping block faces. However, in environmental applications this terrain conformation is not realistic and estimates on the model boundaries could be approximated more precisely.

On the other hand, unstructured grids have the advantage of generality, since they can be made to conform to any desired geometry. For unstructured mesh, the number of cells including a node as vertex is not necessarily the same for all nodes within the domain. The resulting unstructured meshes provide a very powerful tool in discretisation of complex shape. However, unstructured grids require more information to be stored and recovered than structured grids. A popular type of unstructured grid consists of tetrahedral elements. This grid tends to be easier to be generated than others (e.g. hexahedral elements) [Berg et al., 1997]. For these reasons tetrahedral meshes will be mainly used in this work.

The adaptive sampling strategy optimised by uncertainty for unstructured tetrahedral meshes is integrated with the geostatistical analysis on unstructured grids described in Chapter 2.

The diagram that summarises the adaptive sampling with unstructured grids implementation adds some operations, as shown in Figure 4.5 (yellow blocks). In this way it is possible to handle supports with different sizes in the evaluation of the variable distribution.

In the adaptive sampling procedure for unstructured grids, the geometric model of the survey domain is represented with a tetrahedral mesh to better fit the boundaries of the sampling area. In Figure 4.6 an example is shown. From this geometric model, the centroids of each tetrahedral element are computed. They are supplied as input to the SGS to specify the positions where simulations provide the estimates of the environmental variable. SGS returns a value for the estimation of the spatial variable and a value of the uncertainty for each centroid of the tetrahedra of the geometric model. Then, the estimated values are back-transformed using DGM and the change of support coefficients of each cell of the unstructured grid are computed as described in Section 2.2.2. These values are visualised in two maps associating

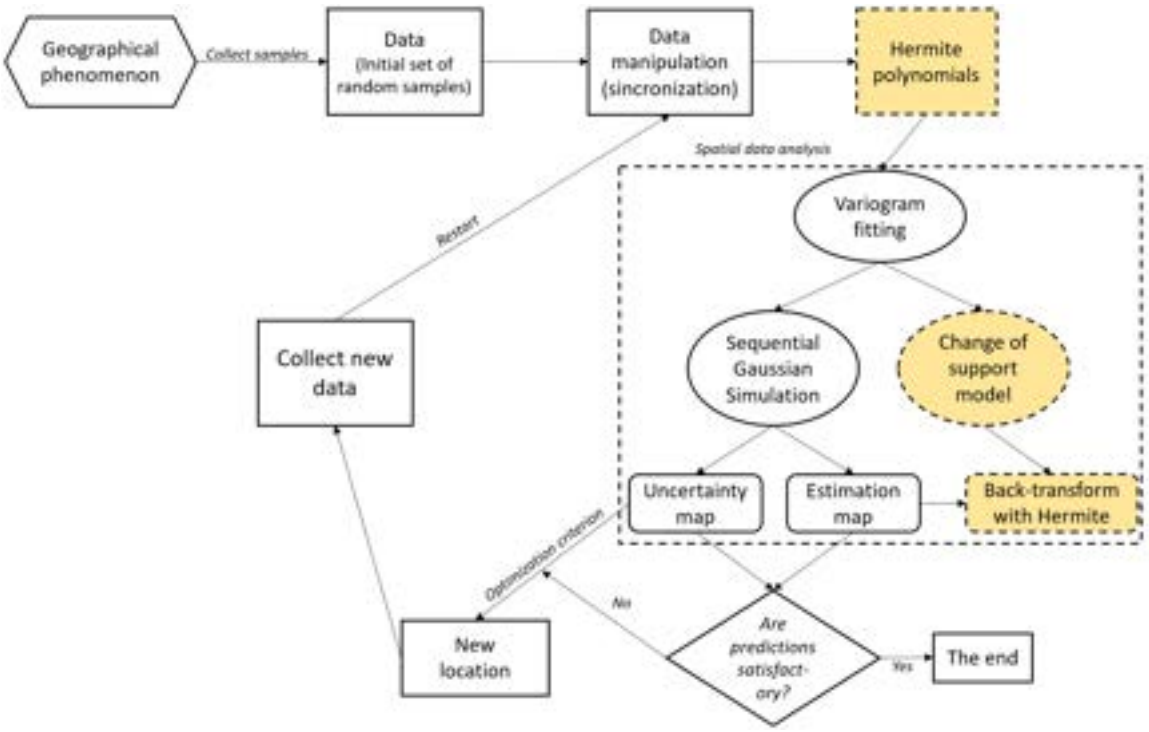


Figure 4.5: Diagram of Adaptive Sampling guided by Uncertainty Minimization on geometric domain represented with unstructured grids.

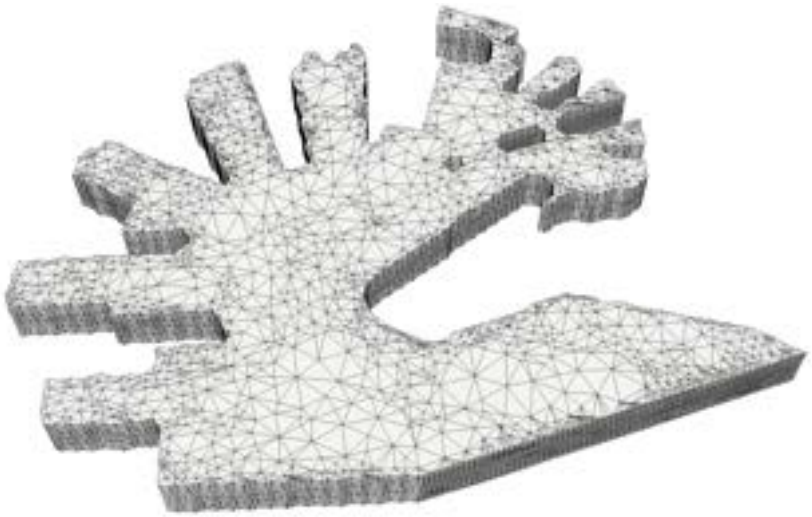


Figure 4.6: Geometric model represented with a tetrahedral mesh of a portion of Genoa harbor.

at each tetrahedron the estimated value and the uncertainty value. The next point to be sampled will be selected at the centroid of the tetrahedron with the highest value of uncertainty.

The whole adaptive sampling strategy is summarised with pseudocode in Appendix B.

Moreover, the unstructured grids allow also to work with problems that may require an adaptive resolution to represent some important areas with a finer resolution and areas with less interest with a more coarse resolution. This adaptive resolution could limit the issue of storing a great number of elements of the grid. This is because when the information about an environmental variable is redundant in a specific area, the resolution of the domain in such area can be reduced. For example, if estimates of a pollutant in an area of the domain are all zero then it is not necessary to represent this area with a huge number of tetrahedral elements, so the resolution could be even more coarse. In this way, the information coming from the map would not be reduced, but computation times and the storage of smaller geometric models lead to a great improvement when the analyses are in situ and in real time. With this approach the resolution of the model could be adapt while the adaptive sampling is proceeding, updating the geometric model of the survey domain at each iteration and simplifying it according to the distribution of the environmental variable.

4.4 Turning the adaptive sampling procedure into an application

A consistent part of the thesis work concerns with the development of the algorithms and code that implement the theoretical framework presented. The value of this implementation work was high as the code represents the tools for running experiments and validating the results. Moreover, a complete graphical user interface was developed to control each phase of the adaptive sampling process and provide visual feedback of the results obtained at each step.

The user interface of the whole process was designed to highlight the interactions and functionalities needed for the main components of the process, using a dialog window divided in four tabs:

1. set-up of a new surveying process - **New Session**

2. set-up the data needed to change support - **Change Of Support**

Figure 4.7: Dialog box for the setting of input data for the simulator of the sampling strategy.

3. provide the visual inspection tools - **Graphical Settings**

4. define parameters for geochemical analysis- **Geochemical Settings**

.

The first tab (Figure 4.7) collects and set-up all the parameters needed to perfom a survey with the approch defined: several inputs are needed, first of all, the geometric model (*model*) of the survey area in which the sampling is carried out.

The second important parameter is the distance threshold (δ) among samples. This parameter is related to a specific configuration of the samples acquisition. Indeed in environmental sampling, data can also be collected during the path to reach the new waypoint with a fixed time interval. δ parameter allows to not consider all the samples positioned too close to each other and to reduce redundancy in the sample size of data. Depending on the experiment, this value can change in magnitude: for example, if the variable to be monitored has low variability and the domain on which it is studied is very large, it would make no sense to sample it every meter, recording the same repeated value. This would also cause problems of clustering. Using the threshold parameter it is possible to store only one value every δ meters, discarding all the others.

During the acquisition campaign, the sampling module will be responsible for data acquisition and communication of it to the calculation module. Conversely, the calculation module will be responsible for communicating the new point to be

Figure 4.8: Dialog box for the setting of parameters for the computation of Change of Support coefficients.

sampled to the sampling module. Both communications will take place by forwarding packets according to the UDP protocol defining with a code through which port this communication occurs.

Other parameters for the adaptive sampling are the number of simulations for the SGS (N_{sim}) and what kind of optimisation the user wants to select (Section 4.2.2) and its corresponding parameters (K and τ). The default is the optimisation by minimising only the uncertainty. An important information is provided about the CPU of the machine in which the software is run in order to know the number of simulations that can be performed in parallel.

In the settings dialog it is possible to define also a folder for the storage of all sampled data and all files that will be generated during the campaign.

Finally, the hardware configuration is shown and the master variable (V_{master}) can be selected as guide of the sampling and for which the uncertainty map will be optimised.

The second tab of the window dialog is about the change of support coefficients computation (Figure 4.8). Here, the method to use for the computation of the block-to-block covariance (Chapter 3) and the corresponding parameters (e.g. the number of points to discretise the tetrahedra) are selected. In the third tab it is possible to specify some graphical settings for the graphs and maps provided to the user during the sampling campaign and in the last tab the geochemical settings can

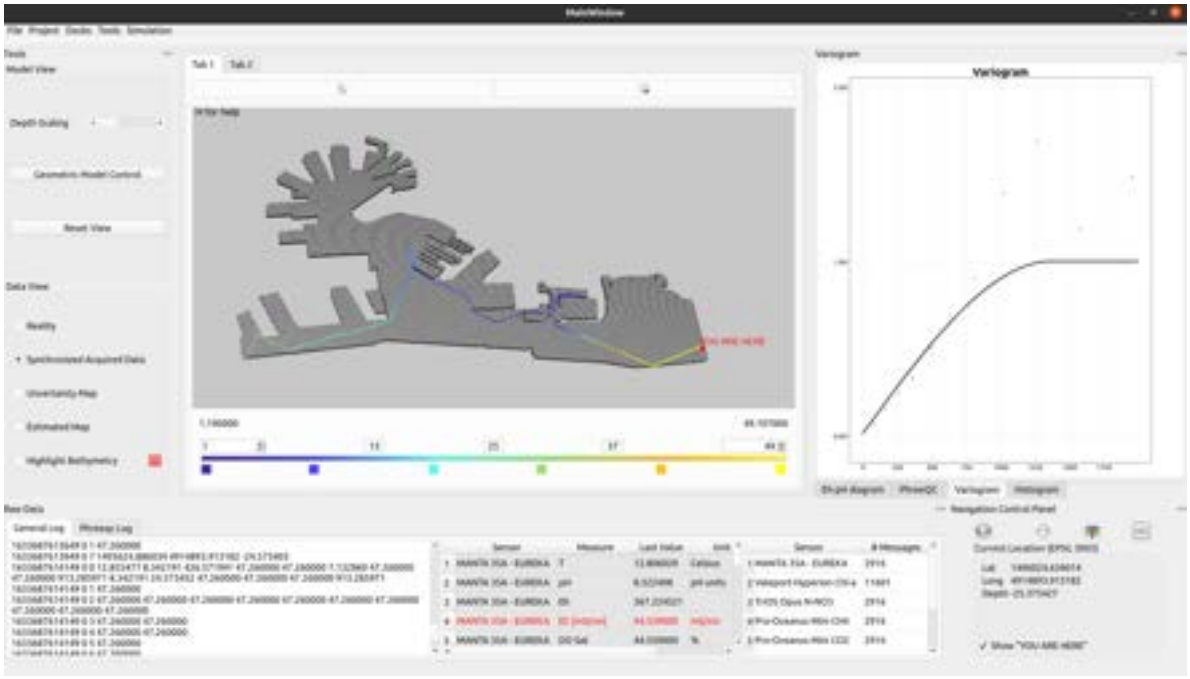


Figure 4.9: Graphical User Interface (GUI) of the application to simulate a sampling campaign.

be adjusted according to the need of the expert. Only the first two tabs are depicted here because they are related to the scope of this thesis.

The Graphical User Interface (GUI) is shown in Figure 4.9. Here the user can visualise the survey domain and monitor results step-by-step. At the center of the GUI there is the geometric model of the survey domain in which can be visualised the displacement of the sampling across the area and also, after the analysis, the estimation map and uncertainty map with their own scales of color (editable).

At the bottom, some of the collected samples are shown to the user to verify and monitor the functioning of the sensors. In the *"General Log"* information about the progress of the estimation process is visualised.

On the right side of the GUI, several graphical tools are shown to monitor the progress of the campaign. This part is composed of several tabs: in the first and second tab it is possible to view some graphs relating to the geochemical analysis; in the third tab the variogram is updated at each iteration of the sampling using all data collected so far; in the fourth tab some histograms of the distributions of the sampled data, transformed data and estimated data are visualised and some comparison among these distributions are shown using the QQ-plots.

In Chapter 5, a simulation of the whole campaign of sampling of a synthetic environmental variable will be presented, monitoring step by step the estimation of the variable distribution and the decreasing in spatial uncertainty using this

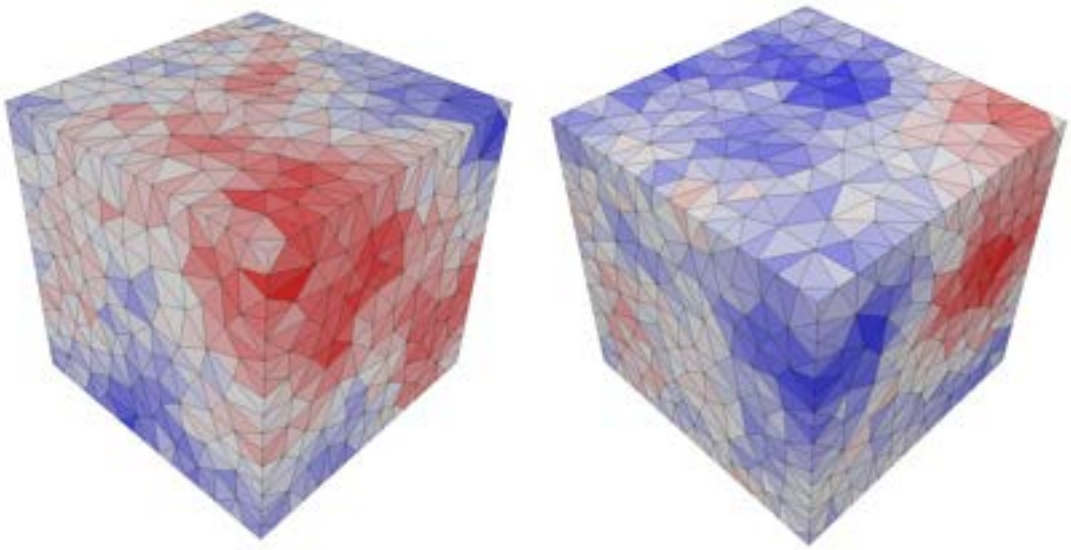


Figure 4.10: Synthetic field for the cubic model generated from an isotropic spherical covariance with a range of 0.8 meters and without nugget effect ($sill = 1$).

application tool.

4.5 Comparison of Sampling Strategies

For the comparison of different sampling strategies with the one proposed in this thesis and to measure the effectiveness of the adaptive procedure, an experiment has been defined.

Two different survey domains are selected for the testing procedure: the first is a geometric model of a cube, $[0, 100]^3$ composed by tetrahedral elements (about 3500 tetrahedra); the second geometric model has a no-convex shape obtained from the subtraction among the cube $[0, 100]^3$ and several parallelepipeds.

Two synthetic scalar fields of hypothetical environmental variable distributions will be evaluated. These two distributions are generated by

- an isotropic spherical covariance with a range of 0.8 meters and without nugget effect (sill equal to one) (Figure 4.10 and 4.12).

- an isotropic spherical covariance with a range of 0.8 meters and with a nugget effect of 0.5 and a sill of 0.5 (Figure 4.11 and 4.13).

The histograms of these synthetic fields are shown in Figure 4.14 and some of the main statistics are in Table 4.1.

The strategies of sampling to compare are:

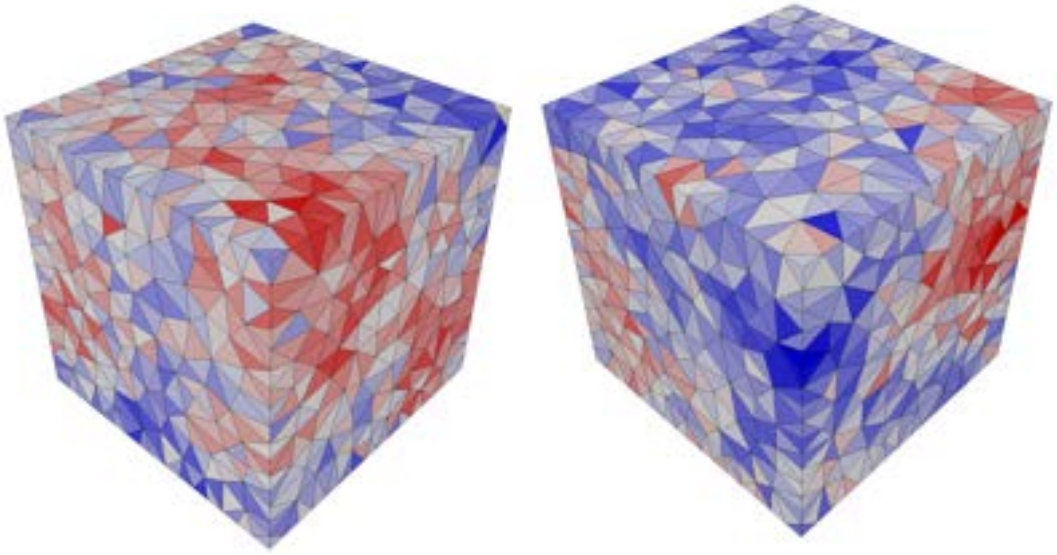


Figure 4.11: Synthetic field for the cubic model generated from an isotropic spherical covariance with a range of 0.8 meters and with a nugget effect of 0.5 ($sill = 0.5$).

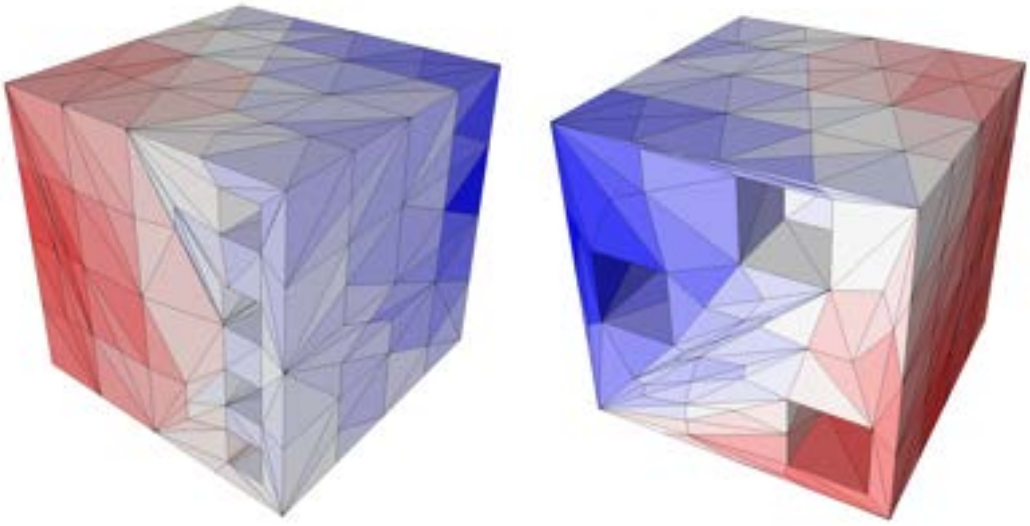


Figure 4.12: Synthetic field for the no-convex model generated from an isotropic spherical covariance with a range of 0.8 meters and without nugget effect ($sill = 1$).

	Min	1st Q	Median	3rd Q	Max	Mean	Variance
Cubic model without nugget	3.190	5.250	5.692	6.079	8.084	5.650	0.441
Cubic model with nugget	4.000	5.400	5.785	6.140	7.414	5.776	0.287
Non-convex model without nugget	2.306	4.086	5.344	5.922	8.435	5.073	1.272

Table 4.1: Summary of distributions of synthetic fields.

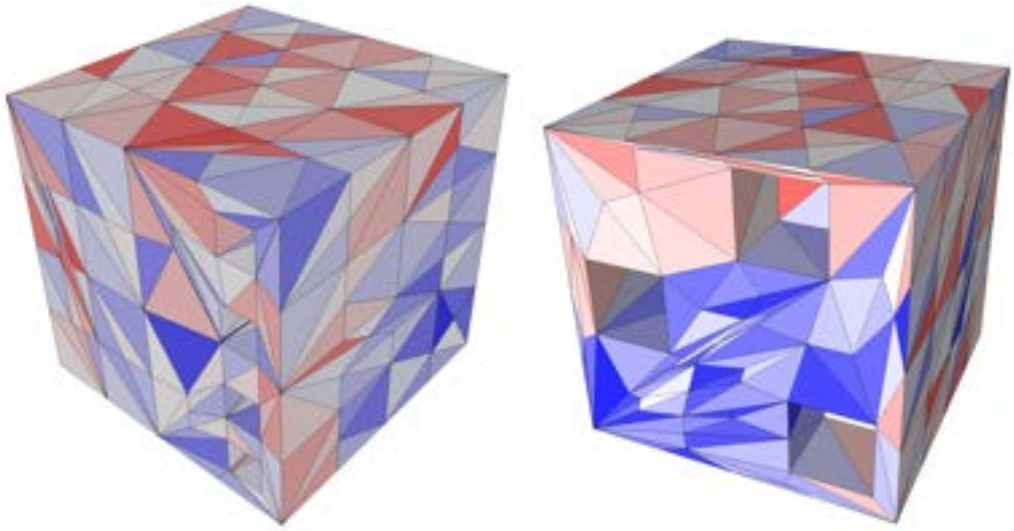


Figure 4.13: Synthetic field for the no-convex model generated from an isotropic spherical covariance with a range of 0.8 meters and with a nugget effect of 0.5 ($sill = 0.5$).

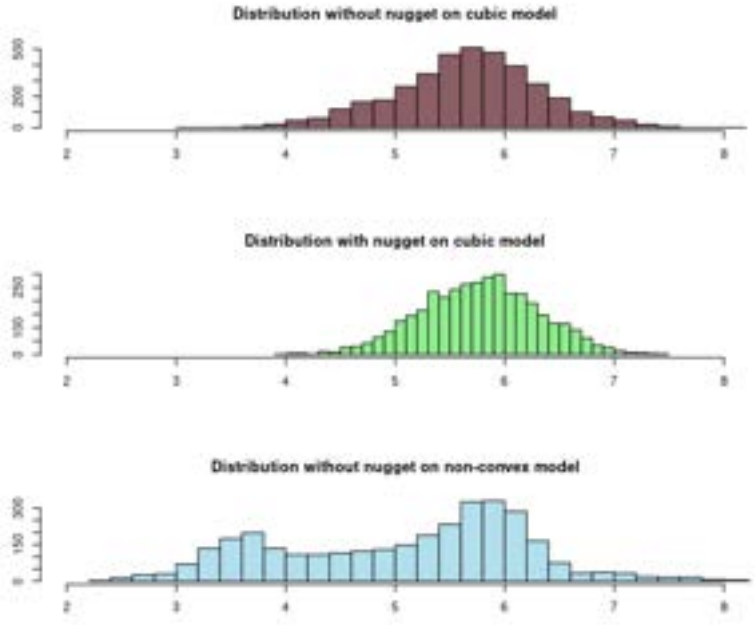


Figure 4.14: Histogram of the distributions of synthetic fields. From top to bottom: synthetic field on cubic model without nugget (Figure 4.10); synthetic field on cubic model with nugget (Figure 4.11); synthetic field on non-convex model without nugget(Figure 4.12).

1. adaptive sampling optimised by maximum uncertainty (AS-U);

2. adaptive sampling optimised by uncertainty and distance (AS-U&D);

3. random sampling (RS);

4. regular sampling (RegS).

In order to measure the reliability of the estimates of each strategy respect to the synthetic field, the Mean Square Error (MSE) is introduced [Hastie et al., 2009]. This allows also to compare results of different strategies identifying the best approach for this experiment.

Especially in environmental sampling the cost associated to the survey has an important role in the selection of the strategy. In this experiment, a function of cost associated with the displacement of the vehicle has been defined. The parameters are the cost of fuel, the fuel consumption of the vehicle per meter traveled and the energy of the battery of sensors. This function returns the cost of the campaign based on the distance traveled. In this way it is possible to show how costs change as the sampling strategy changes, while taking into account the reliability of the estimates of the environmental variable distribution.

In the random strategy the number of samples is N_{tot} and all of them are identified before the start of the campaign and the locations are reached optimising the distance traveled by the vehicle. During this procedure the next point to collect is the closest to the current position: starting from the initial current position (selected randomly) the next point to collect is the closest to that, then the latter position becomes the new current position and the next one is the closest to that and so on, until all samples are visited. In this way the costs associated to the displacement of the vehicle are minimised and the minimal path is selected (Algorithm 3 in Appendix B).

If the adaptive strategy is considered the number of samples N_{tot} is divided into two groups: the number of initial random points (N_{init}) needed to start the adaptive procedure and the number of adaptive points ($N_{adapt} = N_{tot} - N_{init}$) selected sequentially. The initial set of locations is sampled using the minimal path as described above for the random strategy. Then, at each iteration, the adaptive location is reached starting from the current position. This specification could have higher travel costs, but it will be shown that this can be balanced by better estimation results. In order to reduce the cost of travelling, the adaptive sampling optimised by both uncertainty and distance is implemented (Section 4.2.2).

	$N_{tot} = 50$		$N_{tot} = 40$		$N_{tot} = 30$			$N_{tot} = 20$		
	RS	AS-U (20+30)	RS	AS-U (20+20)	RS	AS-U (10+20)	AS-U (20+10)	RS	AS-U (10+10)	AS-U (15+5)
MSE	0.230	0.163	0.256	0.210	0.286	0.261	0.261	0.310	0.298	0.312
COST	12221	36424	10267	26181	9403	21679	15286	6261	12027	9967

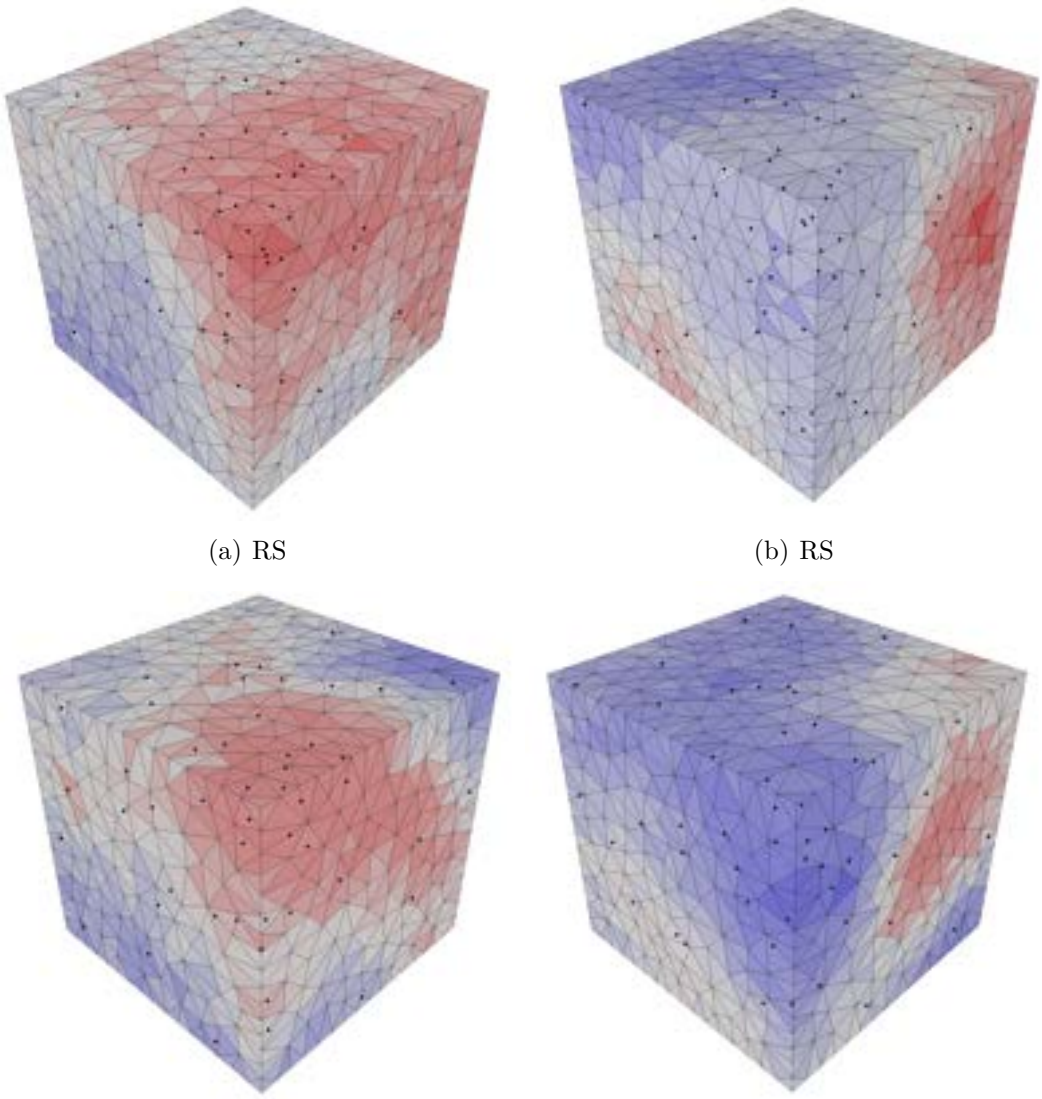
Table 4.2: Mean Square Error and costs of the campaign for different strategy of sampling (random sampling (RS) and Adaptive Sampling with optimisation of Uncertainty (AS-U)), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the model without nugget effect.

Cubic model without nugget effect

The experiment computes the MSE of the estimation map using several strategy of sampling. Starting from a set of samples, geostatistical methods are used to provide the estimation of the environmental variable distribution. In Table 4.2 a summary of the values of MSE for several settings of sampling parameters is reported. In particular, $N_{tot} = 20, 30, 40, 50$ is the number of samples used for the estimation according to the sampling strategy: in the RS approach all N_{tot} points are considered for the computation of the variogram and as input for the SGS; in AS-U approach the total number of samples is divided in a set of points selected randomly and a set of points selected adaptively (N_{init}, N_{adapt}). For example, when $N_{tot} = 50$, the AS-U uses 20 points as initial set to start the iterative procedure and the other 30 points are selected with respect to the uncertainty map as explained in Section 4.2. In Figure 4.15 the estimation maps for the RS and AS-U approaches are shown and they are compared with the synthetic field in Figure 4.12. In this case the MSE is lower for the AS-U (0.163 vs 0.230), indicating better performance in terms of reliability of estimates of the adaptive strategy proposed in this thesis, but the costs associated with the displacement are three times those of the random sampling (36424 vs 12221).

Summarizing, almost all the tests gave the same result: estimates made using the adaptive sampling are more reliable than those obtained through the totally random approach. However, the importance of the costs associated with the campaign should not be underestimated, which in the case of an adaptive sampling are also double or triple those for the random strategy.

To overcome the problem of high sampling costs, adaptive sampling optimised by uncertainty and distance (AS-U&D) is applied: results are shown in Table 4.3. From these results it is evident that, to the detriment of the reliability of the estimates, there is a gain in terms of costs. However, the MSE of RS is bigger then the one of AS-U&D, so the balancing between reliability and costs seems to be the best approach in this case.



(a) RS (b) RS

(c) AS-U (d) AS-U

Figure 4.15: Estimation map for the cubic model of the distribution generated without nugget effect using $N_{tot} = 50$ samples.

	AS-U&D (20+30)	AS-U&D (20+20)	AS-U&D (20+10)	AS-U&D (10+10)
MSE	0.194	0.210	0.268	0.370
COST	26122	18832	13433	10006

Table 4.3: Mean Square Error and costs of the campaign for Adaptive Sampling strategy with optimisation of Uncertainty and Distance (AS-U&D), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the cubic model without nugget effect.

	$N_{tot} = 57$	$N_{tot} = 27$	$N_{tot} = 20$
RegS	0.624	0.344	0.474

Table 4.4: Mean Square Error for the Regular Sampling (RegS), varying the interval among points, for the model without nugget effect.

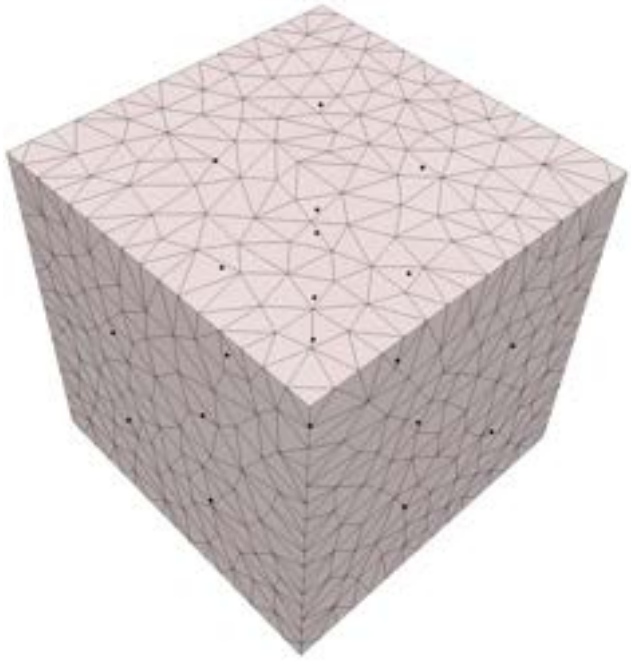


Figure 4.16: Estimation map for the cubic model with Regular Sampling (RegS) of the distribution generated without nugget effect.

Finally, the regular sampling is also tested on this model, obtaining poor results. In Table 4.4 the MSE values are shown. The wide interval between the points does not allow to be able to represent the variations at distances smaller than such interval (e.g. Figure 4.16).

Cubic model with nugget effect

Even when the distribution of the environmental variable presents small-scale discontinuities (nugget effect), it is interesting to study the effectiveness of the adaptive

	$N_{tot} = 50$		$N_{tot} = 40$		$N_{tot} = 30$			$N_{tot} = 20$		
	RS	AS-U (20+30)	RS	AS-U (20+20)	RS	AS-U (10+20)	AS-U (20+10)	RS	AS-U (10+10)	AS-U (15+5)
MSE	0.243	0.224	0.251	0.235	0.262	0.247	0.241	0.291	0.272	0.287
COST	13238	32469	10267	23884	9403	21619	15115	6261	13351	10358

Table 4.5: Mean Square Error and costs of the campaign for different strategy of sampling (random sampling (RS) and Adaptive Sampling with optimisation of Uncertainty (AS-U)), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the model with nugget effect.

	AS-U&D (20+30)	AS-U&D (20+20)	AS-U&D (20+10)	AS-U&D (10+10)
MSE	0.226	0.247	0.248	0.279
COST	24977	20385	13861	9103

Table 4.6: Mean Square Error and costs of the campaign for Adaptive Sampling strategy with optimisation of Uncertainty and Distance (AS-U&D), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the model with nugget effect.

approach to select samples to be used for its estimation.

The experiment computes Mean Square Error of results of the estimation map respect to the synthetic field in Figure 4.11. In Table 4.5 results are summarized.

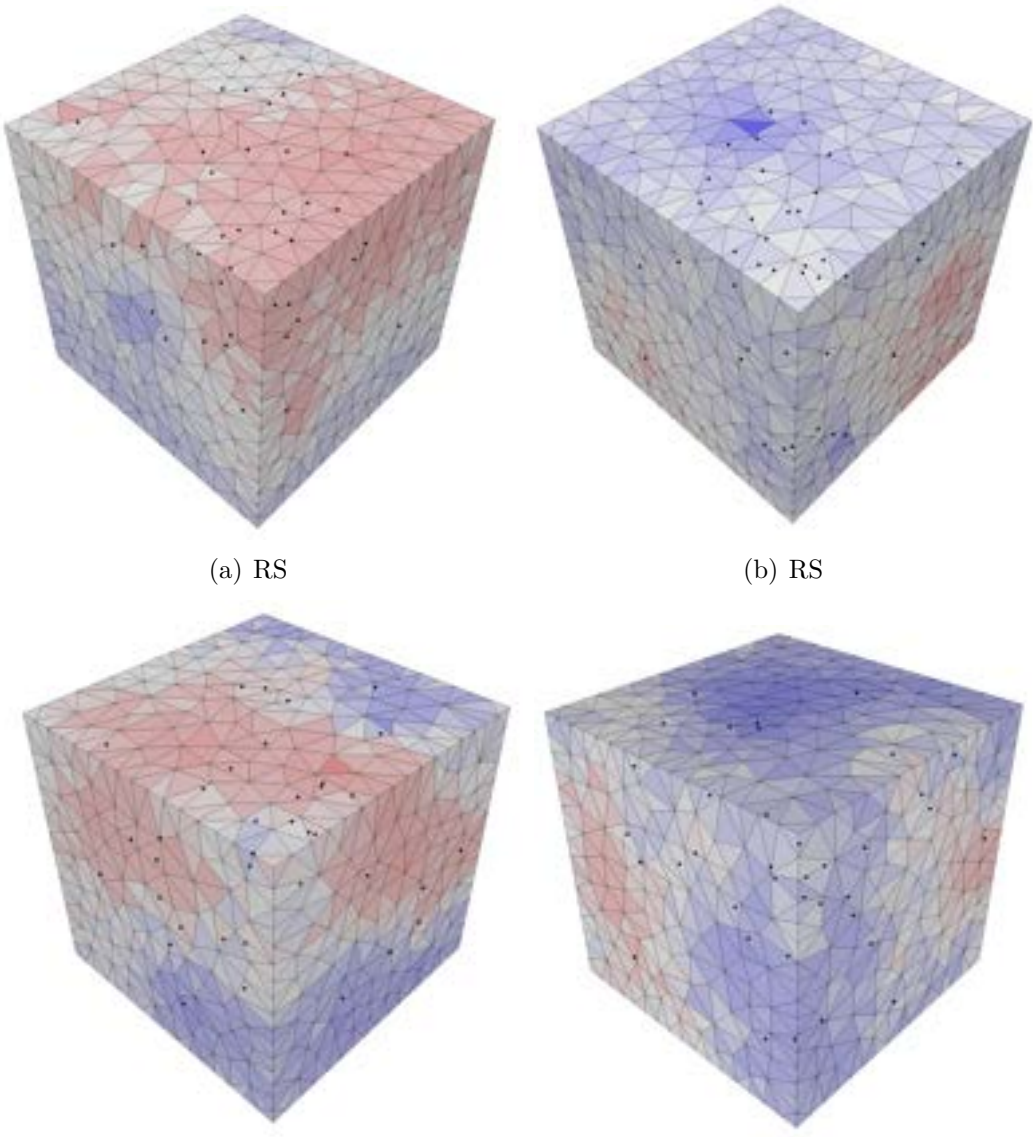
As for the case without nugget effect, the adaptive sampling optimised with the uncertainty has the best performance for the reliability of the estimates with respect to the random selection of samples. For example, with $N_{tot} = 40$ the estimated map of RS and AS-U are shown in Figure 4.17. When the discontinuity is high it is not easy to reproduce the synthetic field with high reliability, but, however, results of the adaptive strategy are better than the ones obtained through other types of sampling.

The cost of the displacement is always higher in the AS-U, since no constraint is required on the distance traveled at each iteration. Again, the AS-U&D is the compromise between being effective and cheap (Table 4.6). The MSE of AS-U&D is higher than the one of AS-U, but the gain in travel costs is evident. However, the performances in the estimation process of AS-U&D are better than the ones of the random approach.

No-Convex model without nugget effect

When a non-convex model is considered, it is interesting to evaluate the displacement costs, since to reach a new waypoint the trajectory can also be very complicated compared to the case of a convex domain.

In this framework, the cost associated to the campaign may be of greater importance in choosing the sampling strategy. Again, the results of the adaptive sampling optimised by uncertainty and distance are the compromise between good reliability in the estimates and lower travel costs. In Table 4.7 a summary of the values of the MSE and costs for several settings of sampling parameters is shown. In particular, varying the number of samples ($N_{tot} = 20, 30, 40, 50$) performances of both random sampling and adaptive sampling optimised by uncertainty are evaluated. In all tests the adaptive strategy provides better results, obviously at a higher cost. For example, for $N_{tot} = 50$ samples the mean square error of AS-U (0.162) is lower than the one of RS (0.219), but the cost of the campaign is double (34421 vs 15871). This



(a) RS (b) RS

(c) AS-U (d) AS-U

Figure 4.17: Estimation map of the distribution generated with nugget effect for the cubic model using $N_{tot} = 40$ samples.

	$N_{tot} = 50$		$N_{tot} = 40$		$N_{tot} = 30$			$N_{tot} = 20$		
	RS	AS-U (20+30)	RS	AS-U (20+20)	RS	AS-U (10+20)	AS-U (20+10)	RS	AS-U (10+10)	AS-U (15+5)
MSE	0.219	0.162	0.253	0.188	0.413	0.289	0.342	1.035	0.461	0.777
COST	15871	34421	14201	26802	10799	21304	19195	8201	14767	12150

Table 4.7: Mean Square Error and costs of the campaign for different strategy of sampling (random sampling (RS) and Adaptive Sampling with optimisation of Uncertainty (AS-U)), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the no-convex model without nugget effect.

	AS-U&D (20+30)	AS-U&D (20+20)	AS-U&D (20+10)	AS-U&D (10+10)
MSE	0.193	0.196	0.344	0.937
COST	21885	17093	12854	10644

Table 4.8: Mean Square Error and costs of the campaign for Adaptive Sampling strategy with optimisation of Uncertainty and Distance (AS-U&D), varying the number of samples, $N_{tot} = 50, 40, 30, 20$, for the no-convex model without nugget effect.

trend is repeated for all the number of samples.

The difference between random and adaptive strategy increases especially when the value of N_{tot} is low (e.g. 20). This is because the randomness of the RS strategy could lead to very particular arrangements of sampling points in space that fail to detect the spatial distribution of the field. On the other hand, the adaptive strategy selects the points to be sampled following an optimisation criterion and, for this, it is not affected by the problem of randomness. In Figure 4.18 the estimation maps of the synthetic field (Figure 4.12) are shown for RS and AS-U strategies with $N_{tot} = 20$ samples.

However, if limiting the costs of the campaign is an important aspect for the selection of the sampling strategy, then, again, the adaptive sampling that optimises both uncertainty and distance traveled is the compromise between a good reliability of the estimates and costs. In Table 4.8 the results of AS-U&D are summarized.

Results obtained using the nugget effect on the no-convex domain are not reported for ease of exposure, since the conclusions are the same of the others frameworks.

4.5.1 Conclusion

In conclusion, the proposed sampling strategy has provided promising results with different types of scalar fields and with different types of domains. This new procedure allows a better representation of the environmental variables of interest with the same number of samples, compared to other types of sampling. In the adaptive strategy, the selection of new samples follows a reasoned criterion of uncertainty

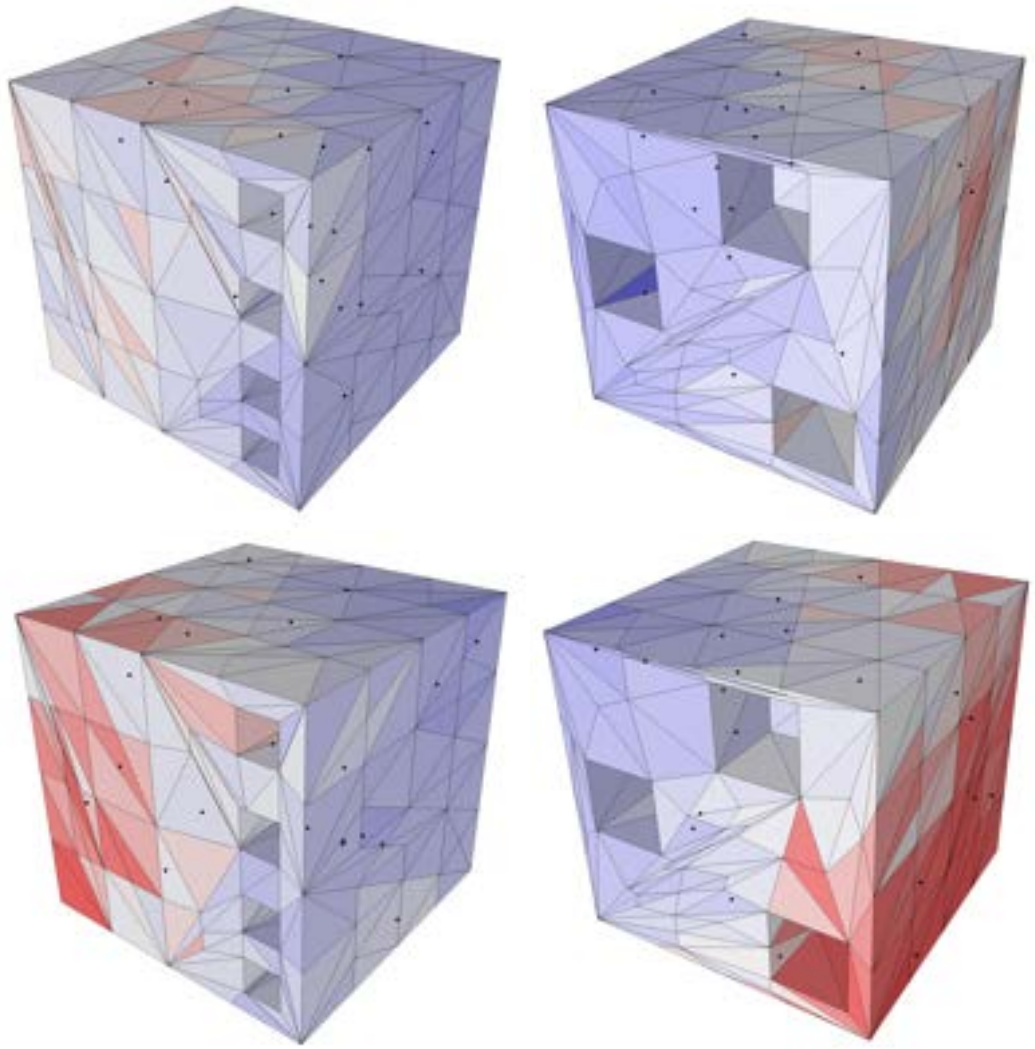


Figure 4.18: Estimation map of the distribution generated without nugget for the non-convex model using $N_{tot} = 20$ samples.

optimisation that avoids wasting resources in sampling unnecessary values.

The biggest flaw of the random strategy is that often the set of selected a priori positions can have a very particular pattern in space (concentration in certain areas leaving other areas completely empty). In these cases the results can be very unreliable.

In the next chapter, the adaptive sampling strategy optimised by uncertainty will be applied on a very complex domain, i.e. the port of Genoa.

Chapter 5

Application to a real case: MATRAC-ACP Project

The method defined for adaptive sampling has been applied and validated in a realistic context where the approach was used to guide a mobile mapping platform for environmental monitoring: this experience was very useful to test the method itself but also to deepen the understanding of how theoretical aspects need to be worked-out to make them viable solutions for innovating traditional practices.

The method has been applied in the context of the project *"Monitoraggio Adattivo in Tempo reale con Automatizzazione del Campionamento - Aree Costiere Portuali - MATRAC-ACP"*, funded by the *EU Interreg VA Italia Francia Marittimo 2014-2020 - Asse prioritario del Programma 2 - Protezione e valorizzazione delle risorse naturali e culturali e gestione dei rischi - Obiettivo specifico della Priorità d'Investimento 6C2-Accrescere la protezione delle acque marine nei porti.* The project was related to monitoring the quality of port waters of the Genoa and Toulon harbours using robotic platforms.

The MATRAC-ACP project provided a perfect context to validate the method also considering the complexity of the harbours' morphology, especially in the Genoa area. The adaptive sampling method has been used to control the movement of the robotic platform: beside the first set of initial samples, the decision on the next sample, and therefore, the next waypoint of the robotic platform was made using the results of the adaptive sampling strategy. The time to reach the new waypoint, or equivalently the distance to it, was an important item to consider as optimisation criteria for the costs associated to the movement of the platform.

This chapter illustrates in details the various steps of the experimentation in the field, some of which are related to the sensor configuration and to the functioning of the measurement they provide. These steps constitute an add-on with respect to

the adaptive strategy defined.

In Section 5.1 the objectives of the project and its general characteristics are presented. Then, Section 5.2 describes how the geometric model of Genoa port has been built and Section 5.3 shows how the collected data is pre-processed to synchronise all samples from several sensors.

Subsequently, the different phases that compose the adaptive strategy and that have been developed specifically for this case study are discussed: the definition of the initial set of samples (Section 5.4.1), the iterative process which chooses the new samples step by step (Section 5.4.2) and finally the stop criterion that defines the end of the campaign (Section 5.4.3).

Then, the integration between the algorithm of the adaptive sampling driven by uncertainty and the change of support model allows to provide the estimation of the environmental variable distribution on unstructured grids. For this, in Section 5.5 a geometric model of a sub-region of the port of Genoa is built with an unstructured grid composed by tetrahedra and a synthetic reality is generated to test the whole iterative procedure.

Finally, in Section 5.6 an application on real data will be presented.

5.1 Project MATRAC-ACP

The MATRAC-ACP project aims to increase the protection of marine waters in ports by improving monitoring procedures through the use of highly automated robotic technologies and adaptive sampling methodologies ([Berretta et al., 2018b], [Berretta et al., 2020]). The aim is therefore to optimise environmental monitoring activities, proposing a new operational method of data acquisition and providing an integrated system consisting of a series of data analysis and real-time visualisation tools.

The introduction of robotics in environmental monitoring procedures guarantees high accuracy and space-time repeatability of measurements. The integration between the calculation software and the visualisation, available to the user during the data collection, allows a high speed of acquisition of data. Furthermore, the adaptive monitoring software provides a real-time planning system of the points to be sampled, with the aim of finding the best relationship between the accuracy of the prediction and the number of samples to be acquired. A modular graphic interface allows the user to view and interpret the environmental situation of the water in real time (during the acquisition campaign).

More concretely, an Autonomous Underwater Vehicle (AUV) is equipped with

sensors that continuously provides environmental data to a processing center. The latter, based on the measurements received, outlines a current map of the distribution of the environmental parameters examined and determines in real time the next movements of the vehicle to acquire further measurements where they are most needed [Caccia et al., 2019].

The AUV is equipped with several sensors; each sensor acquires one or more measurements at a specific frequency, with or without delay times (due to a sensor stabilization or analysis phase), with its own range and unit of measure.

Furthermore, during the displacement of the vehicle samples are acquired also along the path. In this way the number of samples to be processed and analysed at each run does not increase by a single unit, but all the points sampled between the current position and the next point are added to the known data set. This implies that the set of available data that must be analysed in real time can also be very large. This huge data size requires the definition of techniques that are both effective and efficient from a computational time point of view.

In addition, the AUV is equipped with its own GPS for positioning in the survey domain. Using the coordinates provided by the GPS, it is possible to always know the position of the vehicle in space so that it can be maneuvered and displayed on the maps. Moreover, the information on the positioning allows to associate to each measure its position where it was sampled.

Furthermore, the vehicle has a control unit that collects the raw data coming from the sensors and the GPS and prepares them to be communicated to the calculation node by adding a timestamp (an indicator related to the measurement time). The packets are then ready to be sent to the computing node respecting the structure defined by the communication protocol. The calculation node receives the data acquired by the sensors and processes them with geostatistics methods and then communicates to the vehicle the new position of the next point to be sampled. Finally, the distribution of environmental variables and other geochemical parameters are displayed through the user interface.

5.2 Geometric Model of the Genoa Harbour

In the initialisation phase of the sampling system, the 3D model of the domain must be built starting from the available data and supplied as input to the system. The 3D model generated will be the support for the geostatistical analysis operations and for the real-time display of the behavior of the environmental variables of interest.

The available data for the definition of the geometric model of the Genoa's

port comes from two sources: a point cloud representing the bathymetry and a representation of the port infrastructure, which represents the physical boundary of the bathymetry. The point cloud has `XYZ` format and the coordinates are expressed in *Gauss-Boaga*, while the edge of the bathymetry is encoded by a 2D shapefile, according to the Esri standard.

The process of generating the three-dimensional model consists of a preliminary phase of "cleaning" data. The cleaning phase is necessary in order to create a starting dataset without geometric defects, that is, duplicate or almost coincident points, too short edges or similar artefacts. Moreover, the process removes all points whose projection on the plane of the polygon described by the shapefile is not internal to the polygon itself.

Starting from cleaned data, geometric processing operations allow to generate the tetrahedral mesh. The pipeline consists of a first phase in which the surface that encloses the volume of interest is generated, and a second phase of filling the volume itself. The surface will be generated by applying the following procedure:

1. generation of a 2.5D structure (structured grid of triangles) of the bathymetry bound to the edge;

2. projection of the same at sea level;

3. generation of triangular elements that connect the bathymetry and its high duplication (See Figure 5.1).

In the volume filling phase, the tetrahedron mesh is generated using one of the most used algorithms, known as `Tetgen` [Si, 2015].

Beside the tetmesh, also a regular structured 3D grid was also built. The generation of the 3D grid works as follows: first a regular grid representing the bathymetry is built. Then, the grid is "lifted" to the sea level to build the framework to define a regular volumetric grid. The generation of the surfaces that encloses the lower and upper boundary of the volume of interest takes place using the following procedure:

1. generation of structured grid of rectangles or squares of the bathymetry data; the grid defines a 2.5D surface that follows the shape of the sea bottom;

2. construction of a duplicate of the grid, whose node height values are placed at the sea level;

3. generation of geometric elements (rectangles/squares) that connect the sea bottom grid and the sea-level grid.

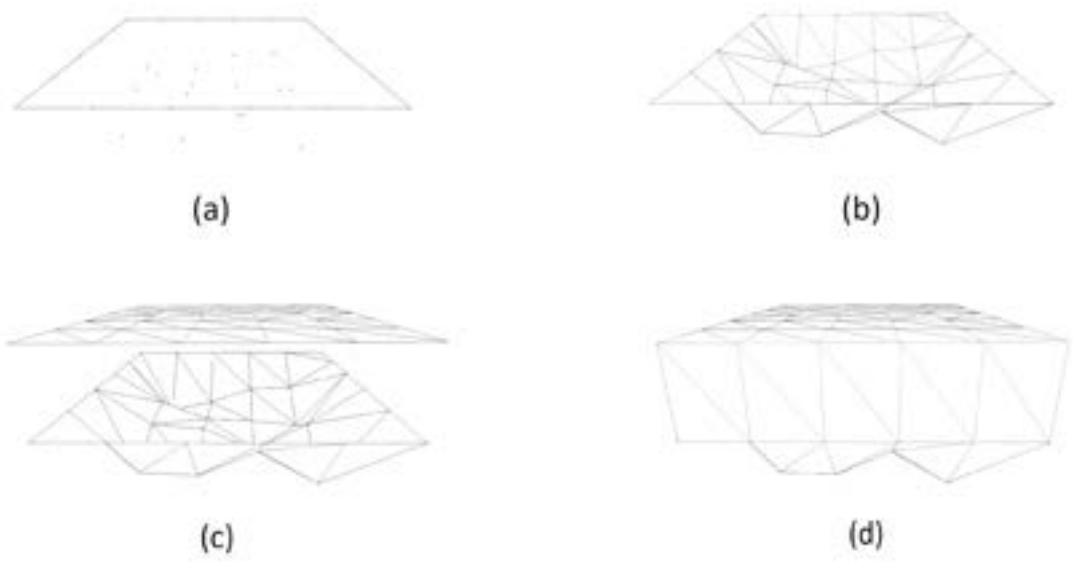


Figure 5.1: The pipeline used for the generation of the tetmesh. (a) Point cloud and shapefile input. (b) 2.5D mesh of triangles which is bound to the edge. (c) Projection of the triangulation at sea level. (d) Surface mesh obtained by connecting the original triangulation with its duplication at sea level.

In the second phase, the 3D grid is constructed by generating hexahedral elements in the volumes obtained by joining each vertex of the sea bottom surface with the corresponding vertex projected at sea level.

The 3D model will be used also to support a digital replica of the process later on, to view and analyze the results of the campaign afterwards.

From a technical point of view, the 3D model generation and saving processes exploit existing geometric processing libraries, with specific reference to `CinoLib` [Livesu, 2019], which provides data structures and geometric editing operations of generic polyhedral meshes.

The tetrahedral mesh and the 3D grid of the Genoa harbour are shown in Figure 5.2 and 5.3, respectively. In a preliminary example we refer to the 3D structured grid as survey domain, while in Section 5.5 the tetrahedral mesh will be considered to show an application of the adaptive sampling strategy on unstructured grids.

5.3 Sensor Data Sincronization

Several sensors are mounted on the robotic platform for measuring various environmental variables. The chemical-physical parameters taken into consideration during the project are mainly temperature (T), salinity based on conductivity with the so

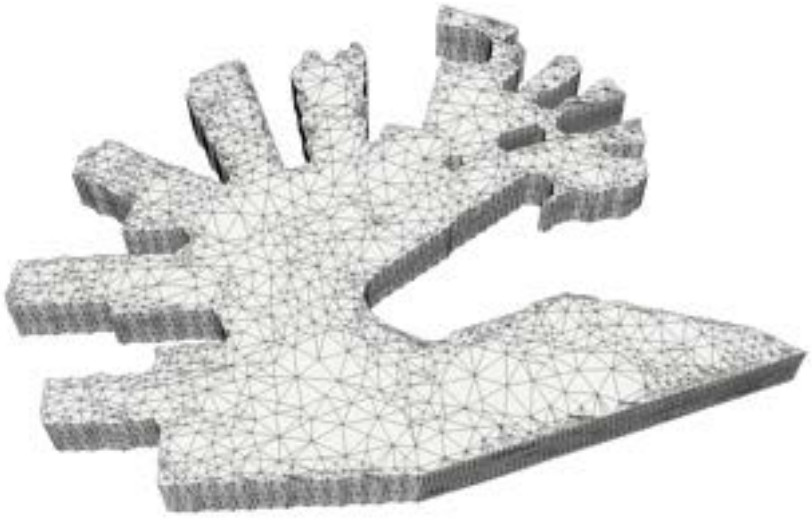
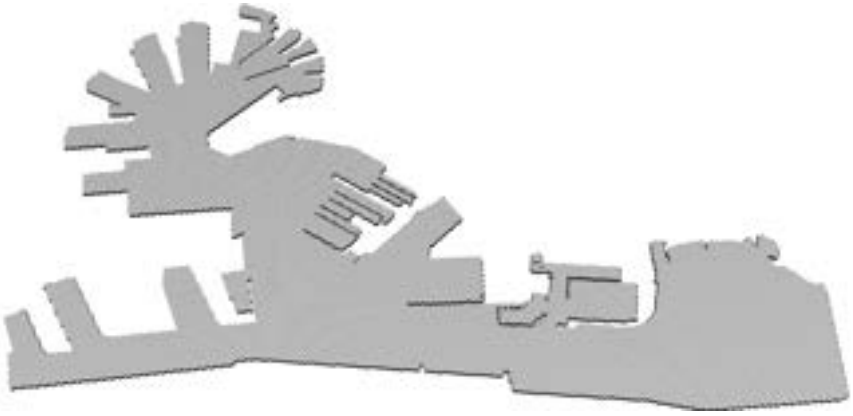
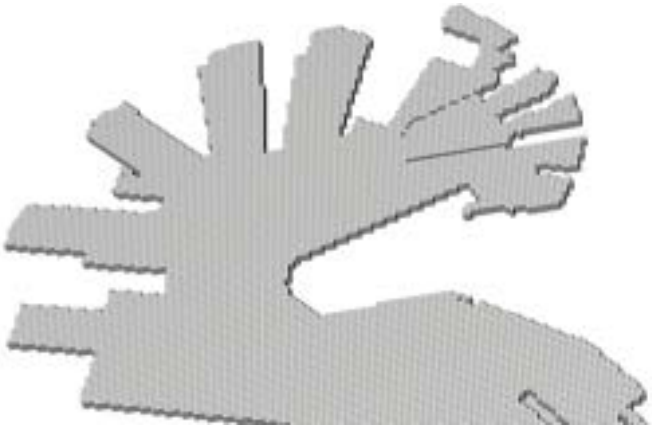


Figure 5.2: The tetrahedral mesh of a section of the Genoa harbour.



(a) Whole water volume of the Genoa harbour



(b) Zoom of the Genoa harbour

Figure 5.3: The 3D grid with regular hexahedral elements of the Genoa harbour.

called Practical Salinity (PSU units), dissolved oxygen (DO in %), pH (pH unit), redox state (mV) and turbidity (NTU unit). Furthermore, the AUV is equipped with a GPS to provide the coordinates of the locations of the vehicle.

Each sensor has its frequency of sampling, that could be different from the others (also the GPS has its own frequency of sampling). Since in spatial data analysis is mandatory to associate a position with each measure, a pre-processing is needed to be able to link data from GPS and data from the other sensors. The aim is to obtain for each position of the GPS the corresponding measures for all sensors.

The timestamp is an integer numeric value that expresses the number of seconds that have elapsed since an arbitrary date, i.e. midnight (UTC) of January 1 1970, a moment which takes the name of epoch. The big advantage of this type of representation of time is that it is easily manageable as it is independent of time zones and allows calculations and comparisons between dates through common mathematical operators. The timestamp allows to identify the moment in which the samples are collected and each sensor and GPS has a timestamp related to their measures.

Moreover, the sensors and GPS can start to samples at different timestamp. In this way, sensors with the same frequency do not necessarily have the same timestamp for their samples.

Consequently, all these problems require a synchronisation procedure that links all the sensors measurements with the GPS positions in order to carry out a well-documented georeferenced analysis.

The structure of sampled data is characterised by a timestamp, an identification code for each sensor and a set of measures whose number (`n_measure`) depends on the type of the sensor. A scheme of this data structure is reported in Table 5.1. In this example, the $ID_{sensor} = 0$ corresponds to the GPS that collects three measures (i.e. the coordinates of locations X, Y, Z). The sensor with $ID_{sensor} = 1$ collects `n_measure` parameters and sensor with $ID_{sensor} = 2$ collects two measures. Therefore, when samples are communicated to the module of sincronisation, a row of data contains a variable number of measurements according to the type of sensor.

Timestamp	ID_{sensor}	Measures
1629899298	1	$\{M1; M2; M3; M4; \ldots; M_{n\_measure}\}$
1629899299	0	$\{X, Y; Z\}$
1629899300	1	$\{M1; M2; M3; M4; \ldots; M_{n\_measure}\}$
1629899301	2	$\{MM1, MM2\}$
1629899301	0	$\{X, Y; Z\}$
...	...	...

Table 5.1: Structure of recorded data from robotic platform.

For the sincronisation, in correspondence with a position sampled by the GPS, the measures linked to that position are obtained for all the sensors that had at least one measure before and after the considered timestamp, by interpolating linearly the values of these measures. Let t_0 be the timestamp of GPS and t_1, t_2 be the two timestamp related to the two measures m_1, m_2 of a sensor. The value corresponding to t_0 is computed as

$$m_0 = m1 + \frac{m_2 - m_1}{t_2 - t_1} * (t_0 - t_1). \tag{5.1}$$

When this is not possible, i.e. there is not a measure before or after the timestamp of the GPS measure, a "Not Available" value is set and is subsequently deleted.

At the end of the sincronization module, data are structured as in Tables 5.2.

Timestamp	X	Y	Z	M1	M2	...	$M_{n\_measure}$
1629899299	x_1	y_1	z_1	$m1_1$	$m2_1$	...	$m_{n\_measure,1}$
1629899301	x_2	y_2	z_2	$m1_2$	$m2_2$	...	$m_{n\_measure,2}$
...	...	...	...	...	...	...	...

Timestamp	X	Y	Z	MM10	MM11
1629899301	x_2	y_2	z_2	$mm10_2$	$mm11_2$
...	...	...	...	...	...

Table 5.2: Structure of data after the sincronization procedure of the sensor with $ID_{sensor} = 1$ (above) and sensor with $ID_{sensor} = 2$ (below).

The synchronisation module is consulted each time that a new set of samples is collected and its result will be the input for the geostatistical analysis.

5.4 Sampling Phases

The validation phase of the proposed method in the context of the MATRAC-ACP project is organised by first defining a simulation scenario to check the entire procedure and its results and then by an experimental phase in the field. Therefore, initially a simulator is exploited to show the phases of the application of the adaptive sampling, subsequently in Section 5.6, the application on real data will be shown.

Once the geometric model is defined, the acquisition campaign can start. Since the uncertainty minimisation of the adaptive strategy considers only one variable at the time, the user must select which is the master variable that will guide the optimised selection of the positions where to sample. The estimated and uncertainty maps displayed to the user will be referred to that variable.

The phases of the adaptive sampling are three: *(i)* collection of an initial set of data; *(ii)* iterative process to select further samples to improve the estimation results; *(iii)* evaluation of the stop criterion that marks the end of the campaign.

To provide a practical example for each step of the adaptive sampling strategy, a simulation of the environmental sampling is developed. In particular, a synthetic reality where to sample the values of the hypothetical environmental variable is defined. Figure 5.4 represents this synthetic field on the geometric model of the Genoa harbour. Furthermore, the simulation approach allows to test the reliability of the results since in this simulation framework the reality that generates the process is known.

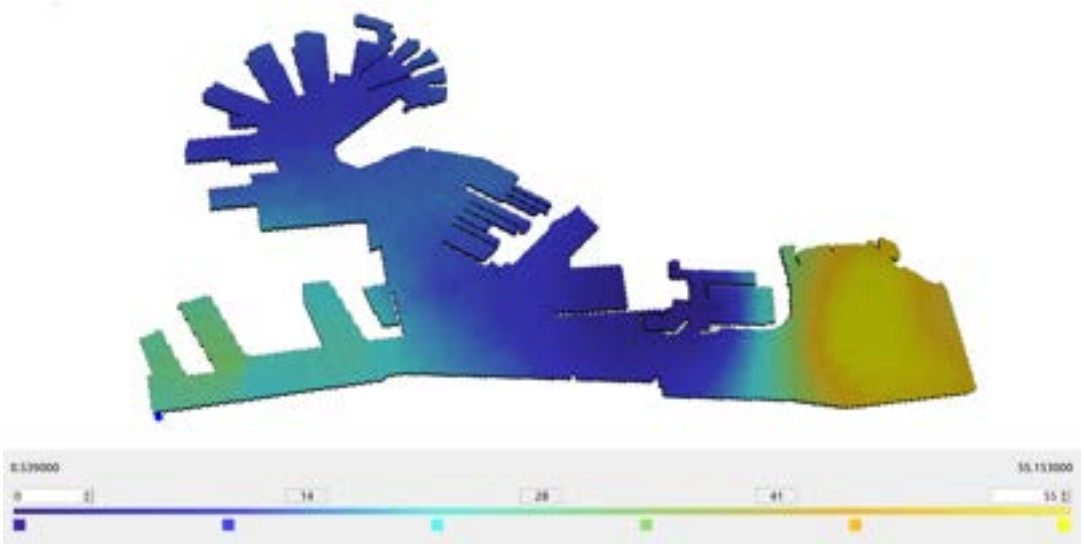


Figure 5.4: The synthetic field of an hypothetical environmental variable on the geometric model of the port of Genoa.

5.4.1 Initial Set of Sample Data

Before starting the survey and choosing the first point to sample based on the optimisation of the uncertainty, it is necessary to collect a set of initial points with which to carry out a preliminary estimate.

The number of initial points ($N_{initial}$) is chosen by the user taking into consideration a priori knowledge of the survey domain, while the positioning of these $N_{initial}$ samples in the volume of water of the port is selected randomly. However, the order in which to visit these positions is not random, but is outlined by an optimised displacement: starting from the current position, the sampling moves to reach the closest location included in the random set of initial points; once this new position is reached, it becomes the current position. The next point to reach is the closest

to the current position and so on until all the measurements in the $N_{initial}$ starting points have been collected. The pseudocode of this phase is in Algorithm 5.4.1 in Appendix B.

The Figure 5.5 shows the displacement of the vehicle that reaches all the $N_{initial} = 5$ locations of the initial set of samples. Then, in this run (Run 0) data is processed (synchronization and geostatistical analysis are applied) and Figure 5.6 shows the preliminary estimation map of the random field of the master variable and its uncertainty map. From these preliminary results the iterative procedure begins and the vehicle moves to reach the next location to sample in the lower right area of the port as indicated by the uncertainty map.

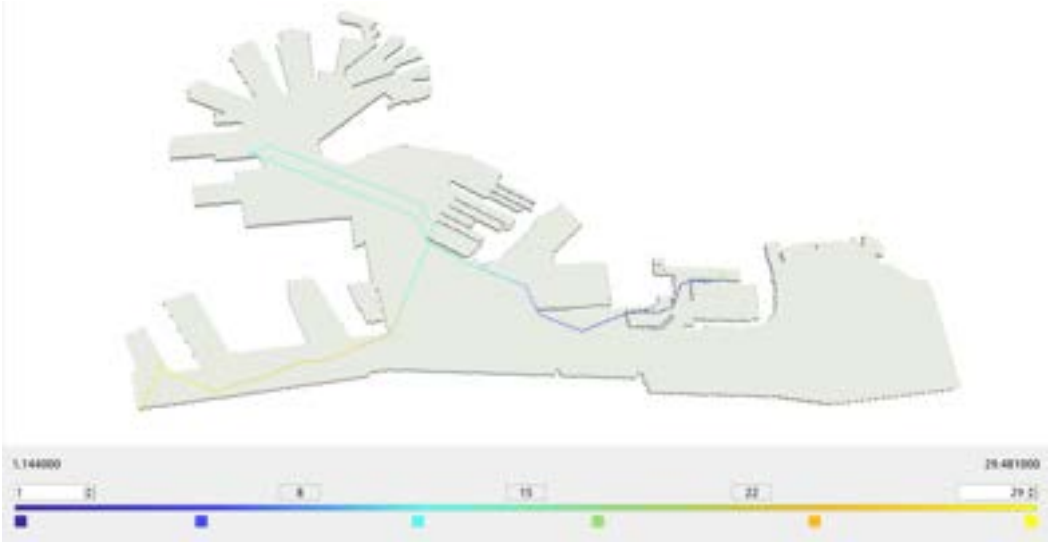
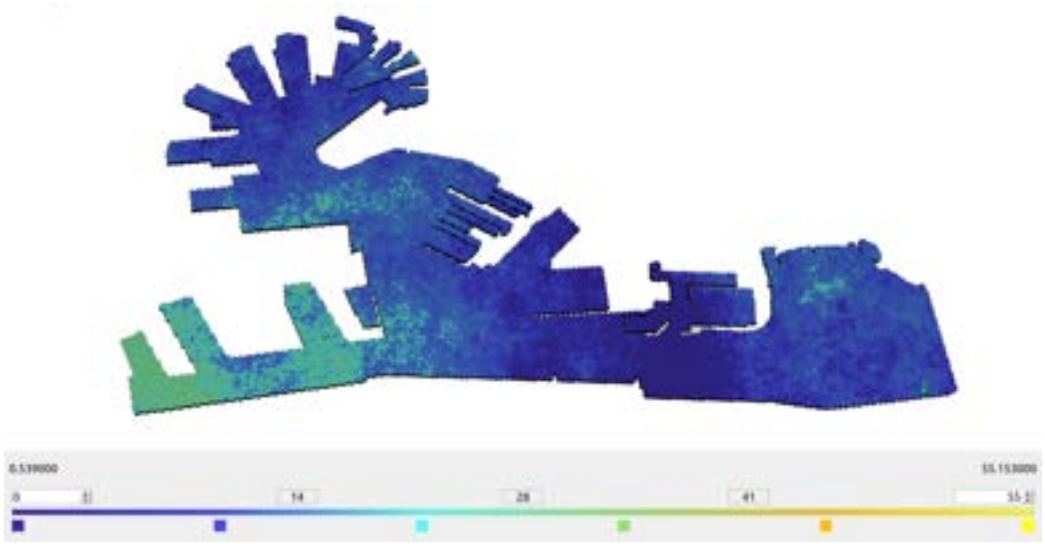


Figure 5.5: Optimal path of the vehicle to reach the five locations of the initial set of samples selected randomly on the survey domain.

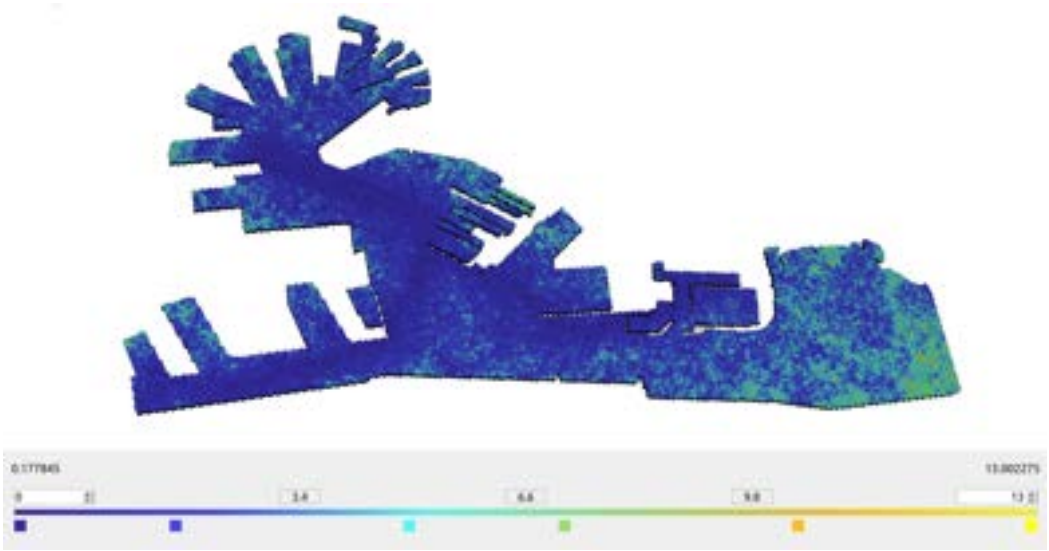
5.4.2 Iterative Procedure

Once the preliminary run is computed from the initial set of data, the iterative procedure starts selecting adaptively the next point to be sampled by optimising the uncertainty map.

The behaviours of the estimation map and uncertainty map during a simulated sampling procedure at different runs are shown in Figure 5.8 and 5.9, respectively. In Figure 5.7 the corresponding variograms are shown: they are computed at each run using data collected so far. The synthetic reality used as reference for this simulation test is generated from a spatial distribution described by an isotropic spherical model with no nugget and with a range of 1500 meters. The computation of the experimental variogram and of its fitting seems to have good results after just



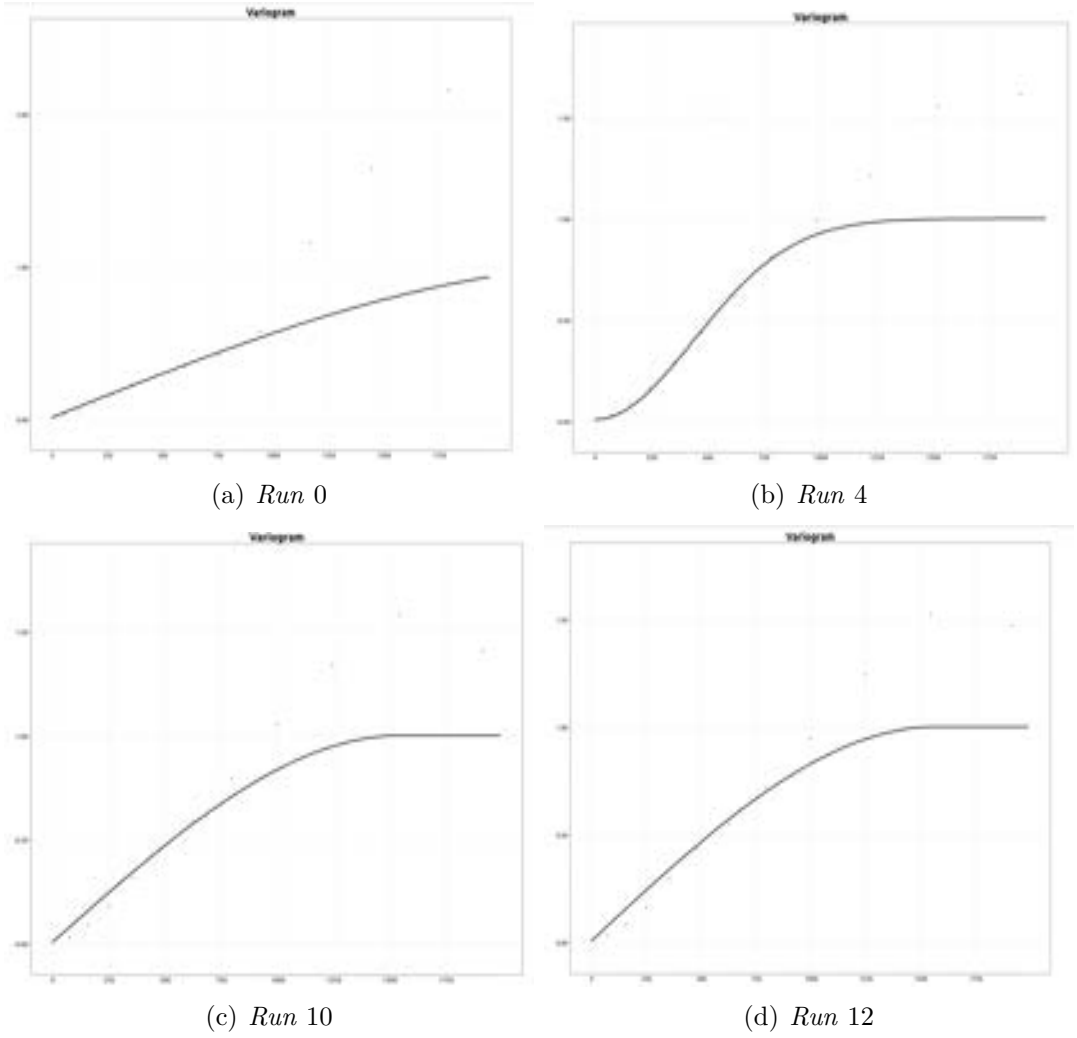
(a) Estimation Map



(b) Uncertainty Map

Figure 5.6: Preliminary estimation map and uncertainty map using the initial set of data for the geostatistical analysis.

few runs. Already after the tenth adaptive sample, the fitted variogram model has a range of 1547 meters and a very low nugget value (0.01). Comparing the estimation maps in Figure 5.8 with the synthetic field in Figure 5.4, results are very close to the "real" values: the area to the right of the survey domain is immediately recognized as the area with the highest values of the variable of interest, while in the upper area where there are the piers of the port the values are the lowest. Finally, on the left side of the port of Genoa, values are estimated as intermediate as they are in the synthetic field. The uncertainty maps (Figure 5.9) are used to determine the next point to be sampled at each run identifying the highest value of uncertainty on



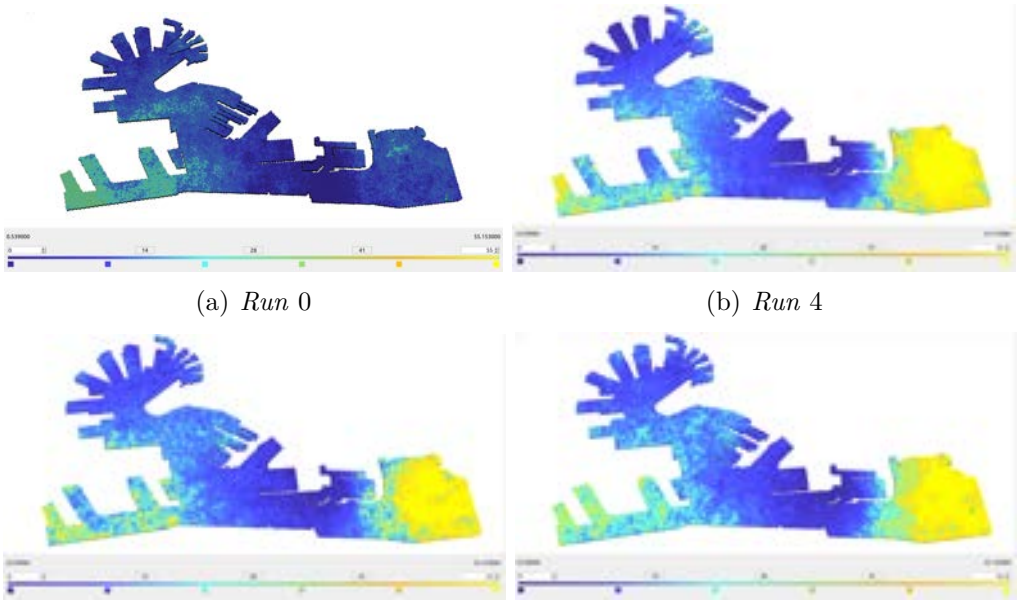
(a) *Run* 0 (b) *Run* 4

(c) *Run* 10 (d) *Run* 12

Figure 5.7: Experimental variogram computed with sampled data up to the i-th run and then fitted with a spatial model: (a) $type = Sph$, $range = 2543$, $sill = 0.99$ and $nugget = 0.01$; (b) $type = Gau$, $range = 617$, $sill = 0.99$ and $nugget = 0.01$; (c) $type = Sph$, $range = 1547$, $sill = 0.99$ and $nugget = 0.01$; (d) $type = Sph$, $range = 1565$, $sill = 0.99$ and $nugget = 0.01$.

the map. In this example only the uncertainty is optimised and the displacement of the vehicle is not considered in the criterion of optimisation.

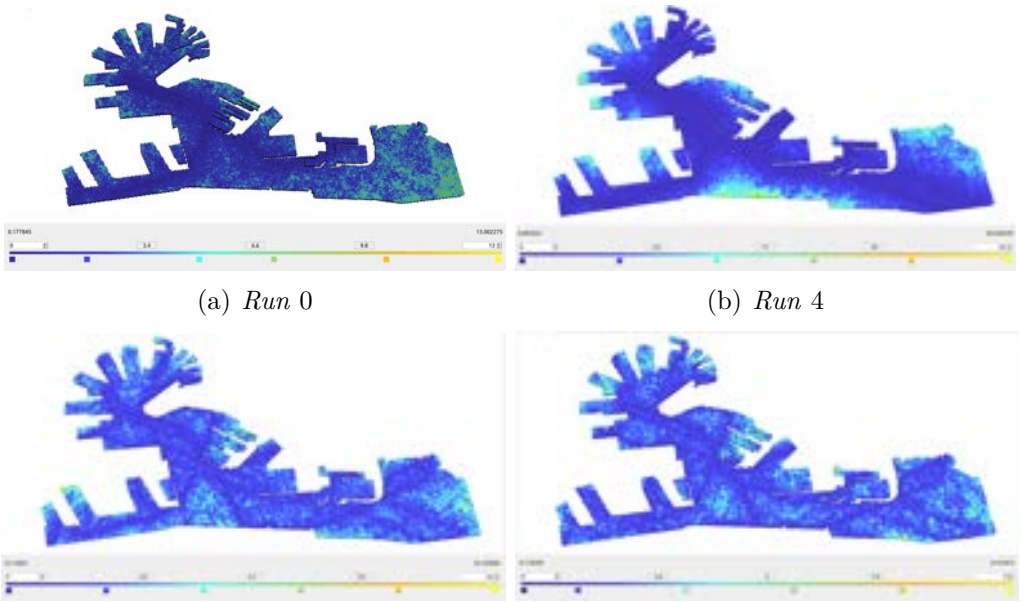
Moreover, the values of uncertainty of each node of the 3D grid is averaged to provide a global measure of uncertainty: Figure 5.10 shows the decreasing trend of this global uncertainty as samples are added to the available data. This iterative procedure continues until a stop criterion is reached.



(a) *Run* 0 (b) *Run* 4

(c) *Run* 10 (d) *Run* 12

Figure 5.8: Estimation maps of the environmental variable at different runs of the algorithm.



(a) *Run* 0 (b) *Run* 4

(c) *Run* 10 (d) *Run* 12

Figure 5.9: Uncertainty maps of the environmental variable at different runs of the algorithm.

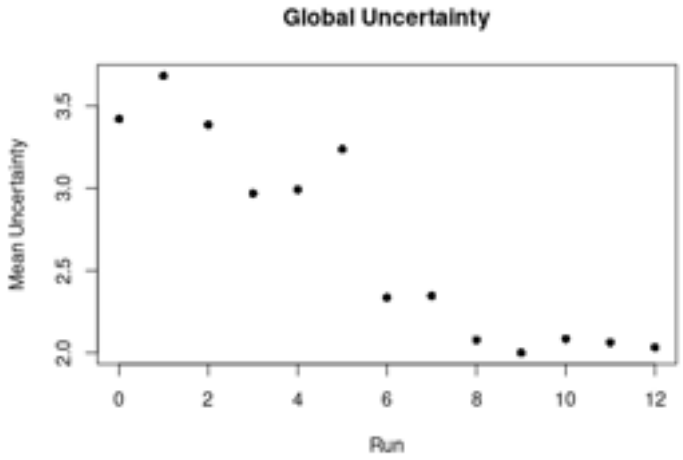


Figure 5.10: Behaviour of the spatial uncertainty summarised with a mean-uncertainty value at different runs.

5.4.3 Stop Criterion

The stop criterion defines when the expert is satisfied with the results and the whole campaign ends. This may be a user's choice or an absolute measure computed automatically by the system.

The criterion can be based on the global uncertainty as in the Figure 5.10: when there is no longer a significant improvement in reducing uncertainty then sampling stops. For example, in the graph it can be seen that after the ninth run the reduction of global uncertainty is no longer evident and the addition of new samples does not provide a visible improvement. So, at *Run* 12 it was decided to stop the acquisition.

5.5 Application on Unstructured Grid

When the geometric model is built with a tetrahedral mesh as in Figure 5.2 and its elements are tetrahedra with different sizes the change of support model is a very effective methodology to provide a reliable estimation of the enviromental scalar field associated to the unstructured grid, as shown in Chapter 2.

For a simulation test, a synthetic reality is related to the unstructured grid in which to sample the values of the hypothetical environmental variable. Figure 5.11 shows this synthetic field on the geometric model of a sub-region of the Genoa harbour. The synthetic reality is generated from a spatial distribution described by an isotropic spherical model with a moderate nugget (20% of the total variance) and with a range of 600 meters. The computation of the experimental variogram and of its fitting seems to have good results after just few runs. Already after the fifth

adaptive sample, the fitted variogram model has a range of 604 meters and a nugget value of 0.13 ($sill = 0.87$) as shown in Figure 5.12.

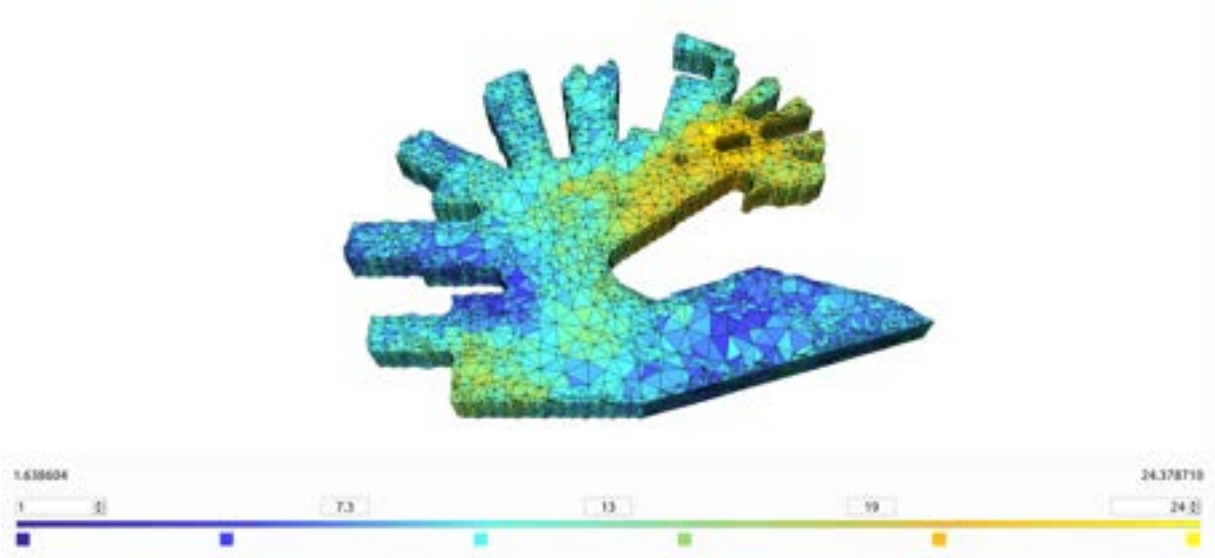


Figure 5.11: Synthetic reality on unstructured grid.

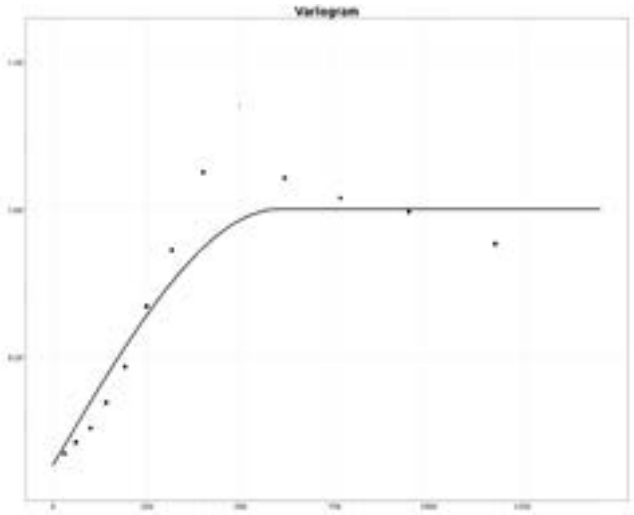


Figure 5.12: Variogram at *Run* 5. Parameters: $range = 604$ meters, $nugget = 0.13$ and $sill = 0.87$.

The adaptive sampling strategy starts with the selection of five random points as the initial set of data. In Figure 5.13 the path to reach these locations is shown.

The preliminary estimation map and uncertainty map deriving from the analysis of this set of data are in Figure 5.14 (a) and 5.15 (a). This estimation map has little variability and is unable, considering only samples collected so far, to provide

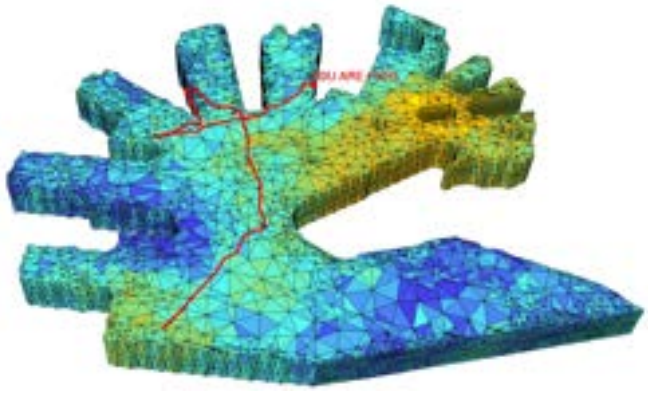


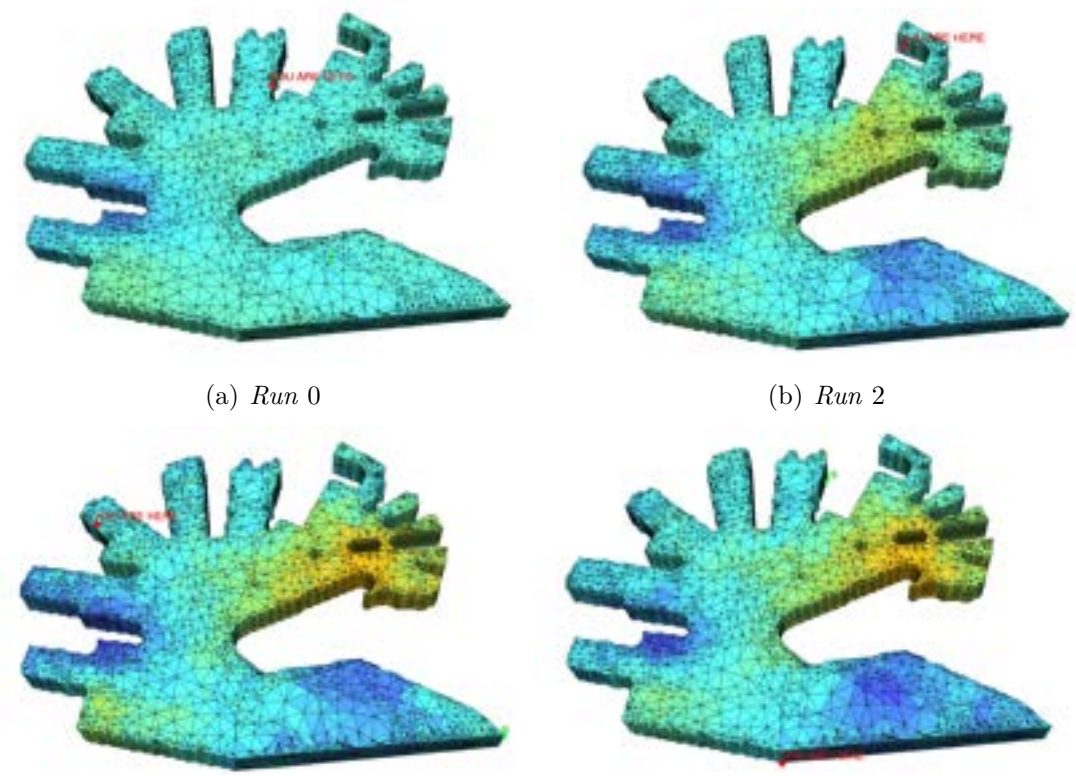
Figure 5.13: Path of the vehicle to reach the five samples of the initial set.

satisfactory results. For this, new samples are needed to improve the estimation of the scalar field. Looking at the uncertainty map, the next point to be sampled can be identified in the lower region of the map. This marks the beginning of the iterative process of the adaptive sampling driven by uncertainty. Already with the fifth sample selected adaptively it can be seen how much the results of the estimation are very close to the synthetic reality from which the distribution is generated. Indeed, comparing Figure 5.11 and Figure 5.14 (c) the difference between the two estimation maps is very low. However, the sampling strategy ends at the ninth sample, when the decrease of spatial uncertainty is no longer significant compared to the effort and costs of adding new data.

In this estimation procedure the traditional geostatistical analyses is integrated with the Discrete Gaussian Model, in particular the DGM-2 (Section 2.2.3). The change of support coefficients (r_i) are computed at each step of the sampling procedure by exploiting the fitted variogram parameters and by discretising each tetrahedron of the geometric model with the Sobol's algorithm. In general, a high value of nugget is reflected in a change of support coefficient further from 1. In this case, the nugget value is estimated approximately equal to 0.20 and the distribution of the coefficients of the cells at a specific run (*Run* 5) is shown in Figure 5.16.

In conclusion, simulation tests have provided very interesting results. The adaptive selection of locations plays a crucial role when the environmental surveys must comply with certain specifications such as shorter times and lower costs. Our proposal provides an algorithm that works in real time to establish the state of health of the environment that has been considered. This allows to make decisions quickly without having to wait for long laboratory analyses and avoid issues related to waiting for results.

Furthermore, the introduction of unstructured grids has allowed the representa-



(a) *Run* 0 (b) *Run* 2

(c) *Run* 5 (d) *Run* 9

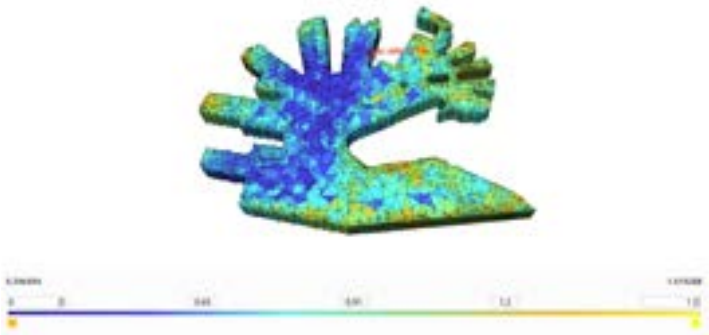
Figure 5.14: Estimation maps of the environmental variable distribution at different runs of the algorithm.

tion of even very complex domains with high precision. The extension of change of support models to unstructured grids made it possible to estimate efficiently the spatial distribution even if supports have different sizes.

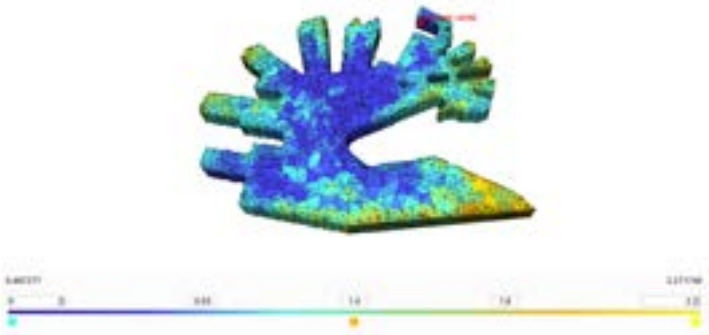
5.6 Application on Real Data

In this section a real application of the strategy of the adaptive sampling with optimisation of uncertainty will be described.

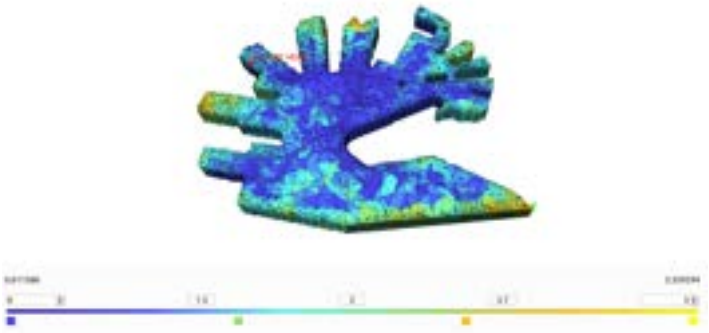
In this real application the survey domain is the port of Genoa where the area on the right is off-limits to navigation as it is shown in Figure 5.18. For this reason, even when the uncertainty map indicates to move the sampling in that area (based on the optimisation criteria), it will be ignored and the vehicle will move to the point of maximum uncertainty within the zones where it is allowed to navigate. There are two possibilities to treat areas of the geometric model when they are off-limits: *(i)* it would be possible to assign weights to the cells of the grid and set to zero those that cannot be reached by navigation; *(ii)* build an *ad hoc* model for the campaign



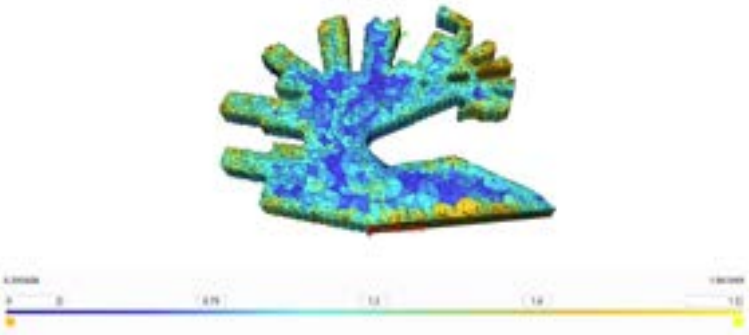
(a) *Run* 0



(b) *Run* 2



(c) *Run* 5



(d) *Run* 9

Figure 5.15: Uncertainty maps from the SGS at different runs of the algorithm.

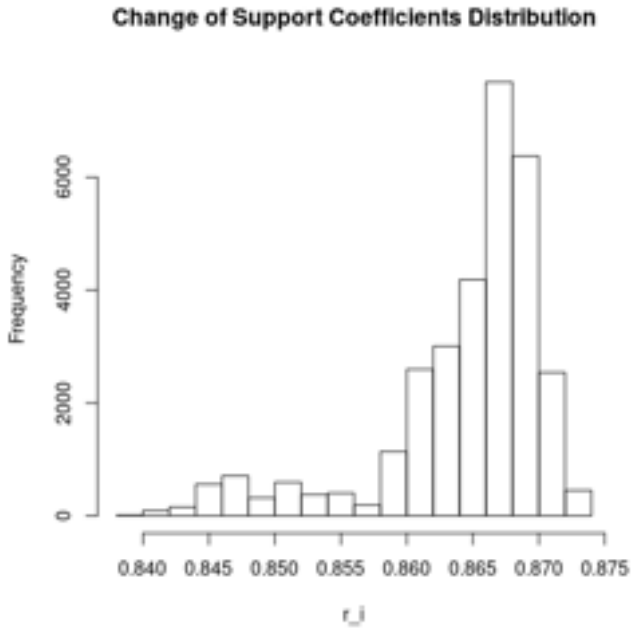


Figure 5.16: Change of support coefficients distribution at *Run* 5 using the fitted variogram in Figure 5.12 and Sobol's algorithm for the discretisation of tetrahedra.

by eliminating prohibited areas. In our application we use the configuration *(i)*.

In this survey the vehicle is mounted with several sensors for measuring the environmental variables (Figure 5.17). The master variable in this case is the percentage of Dissolved Oxygen and the study is concerned to estimate its spatial distribution.

Figure 5.17: Robotic platform mounted with several sensors used in MATRAC-ACP project.

In the port areas there are several constraints and laws for navigation. For this, the autonomous vehicle is not free to move but it must be followed by a boat on which the computer that receives the data is transported. On this boat the expert can evaluate results in real time through the graphic interface that has been developed.

After some initial random acquisitions, necessary to start the iterative procedure,

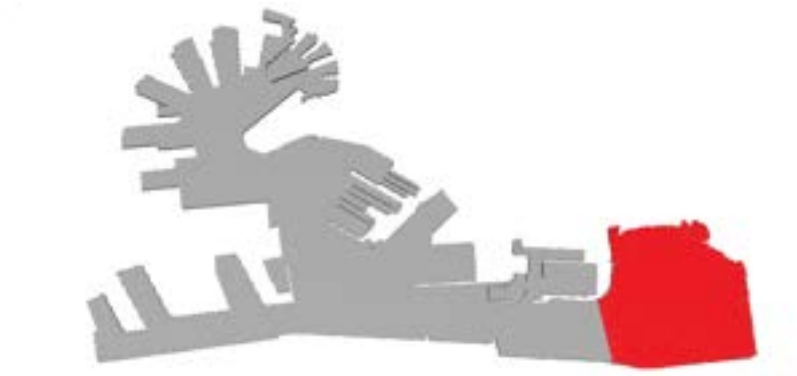
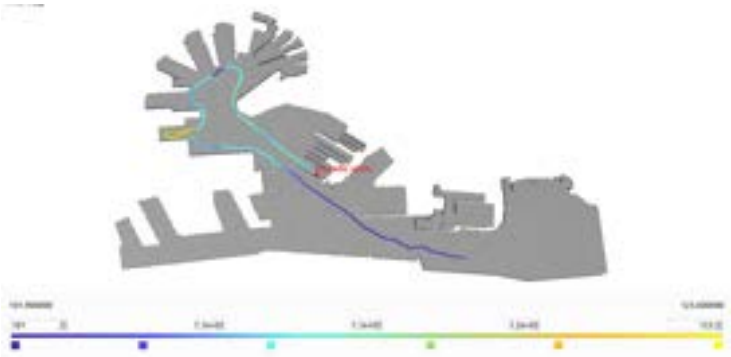


Figure 5.18: Navigable survey domain of the port of Genoa. The red area represents the no-navigation zone.

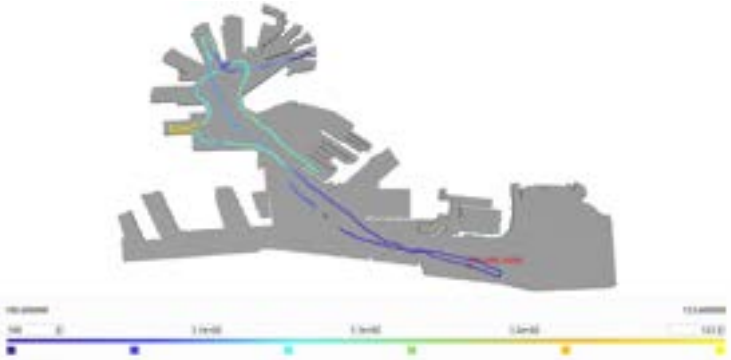
the sampling strategy proposed in this thesis has begun. Figure 5.19 shows the steps of the displacement of the vehicle during the survey. In this campaign, four runs have been executed before ending the acquisition.

The variogram at each step (Figure 5.20) is always available to the expert to monitor the acquisition on the graphic interface of our software. Using these fitted variograms the software provides the estimation map (Figure 5.21) and also the uncertainty map (Figure 5.22) where the next location is selected at each step.

In a real application the spatial distribution of the environmental variable is not known and comparison to evaluate results of the estimates is not possible. However, the expert confirms a coherent behaviour of the estimated values of Dissolved Oxygen.



(a) *Step* 1

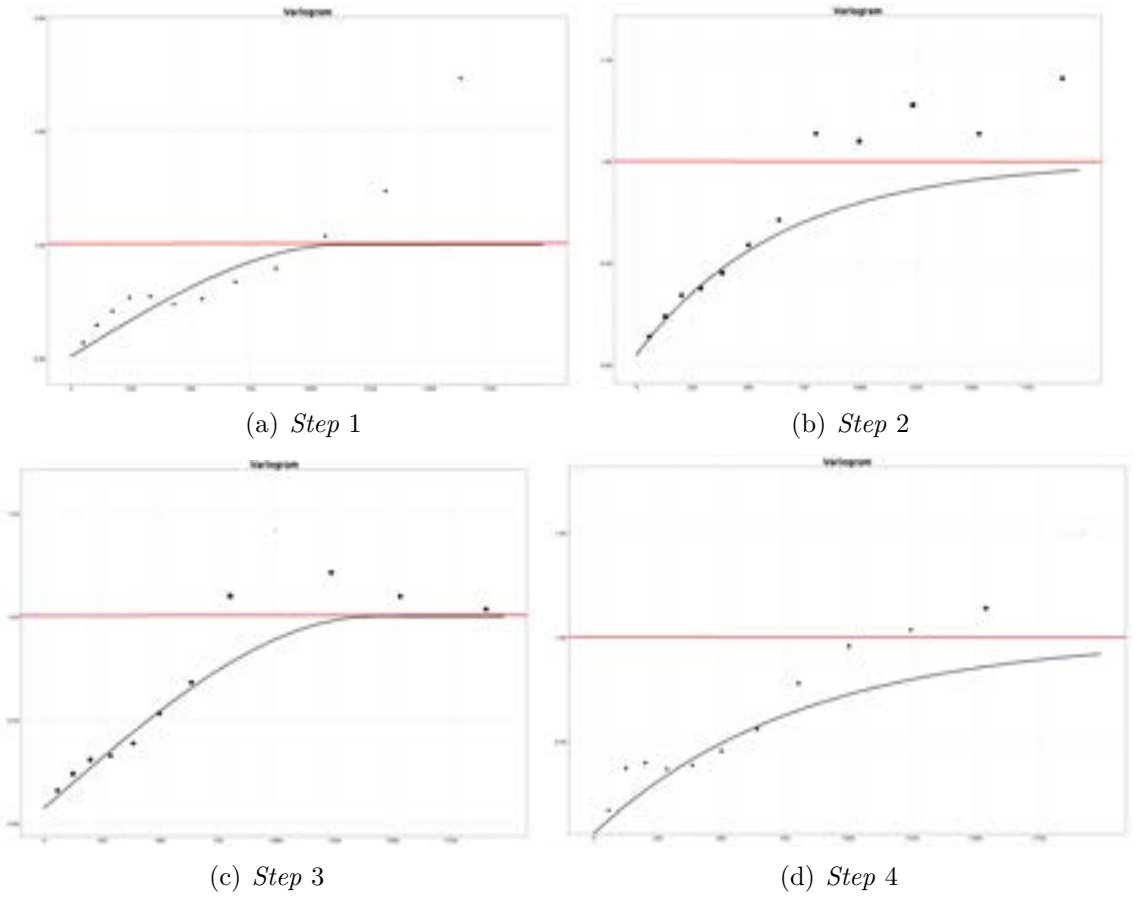


(b) *Step* 2

(c) *Step* 3

(d) *Step* 4

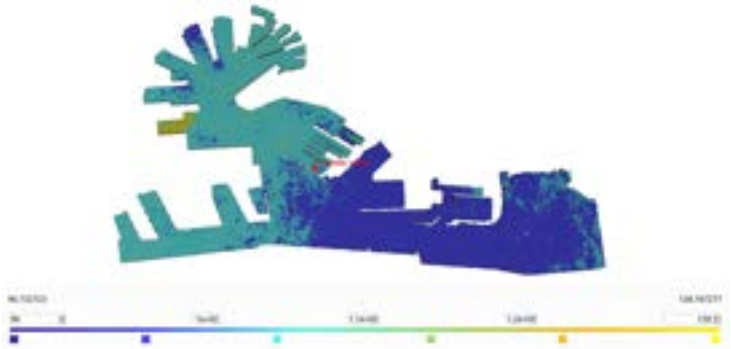
Figure 5.19: Paths between pairs of sampled points in Genoa port.



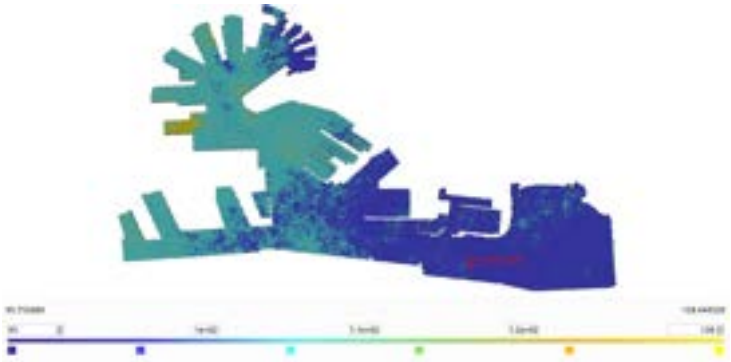
(a) *Step* 1 (b) *Step* 2

(c) *Step* 3 (d) *Step* 4

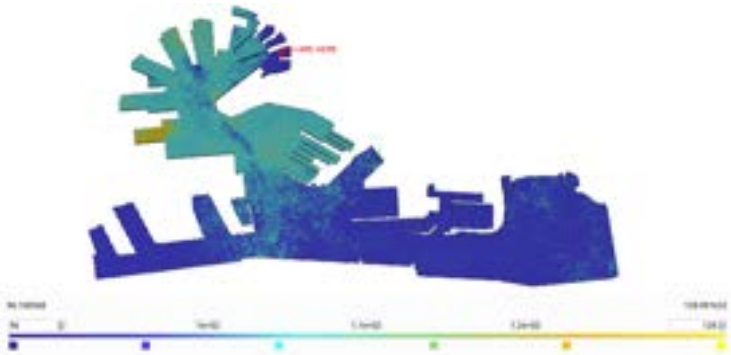
Figure 5.20: Variogram fitting at each step of the sampling. *Step* 1: Spherical model with no nugget variance and a range of about 1000 meters; *Step* 2: a spherical model with a low value of nugget and a range of about 1750 meters; *Step* 3: a spherical model with a low value of nugget and a range of about 1200 meters; *Step* 4: a spherical model with no nugget variance and a range of about 1750 meters.



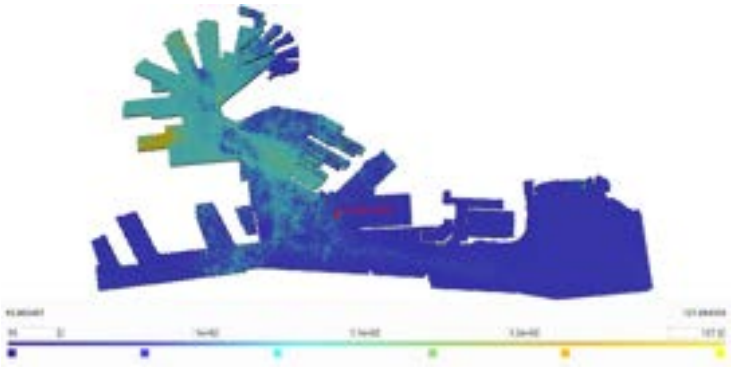
(a) *Step* 1



(b) *Step* 2

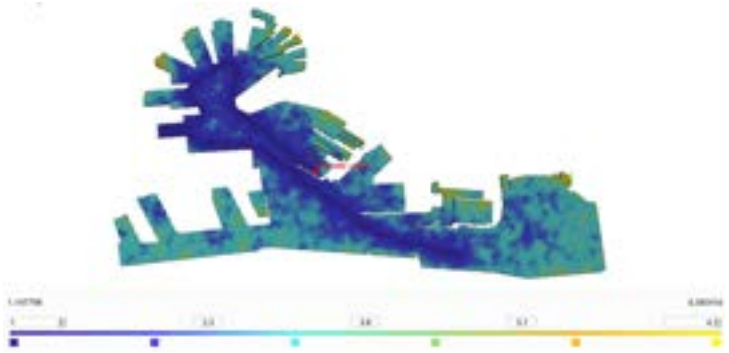


(c) *Step* 3

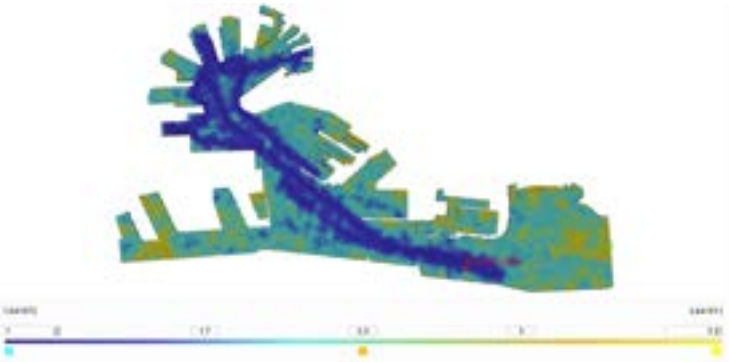


(d) *Step* 4

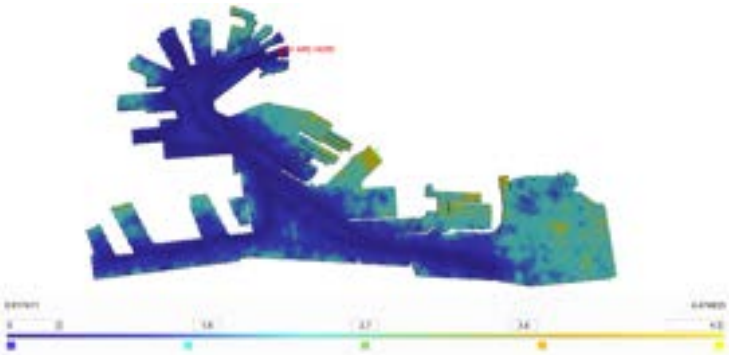
Figure 5.21: Estimation map at each iteration of the sampling in Genoa port.



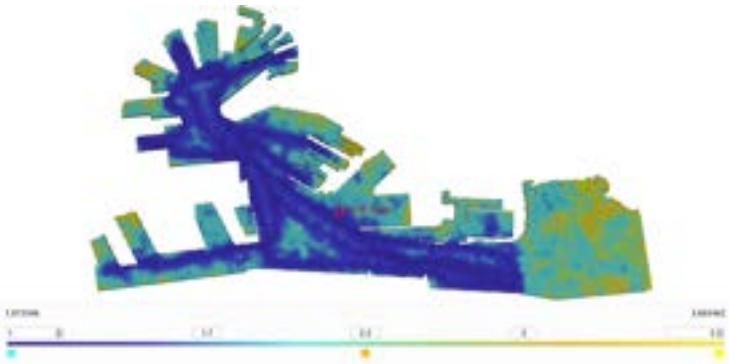
(a) *Step* 1



(b) *Step* 2



(c) *Step* 3



(d) *Step* 4

Figure 5.22: Uncertainty map at each iteration of the sampling in Genoa port.

Conclusion

The research program addressed during the thesis entailed the study of spatial data analysis and the development of a mathematical framework able to support a novel and effective sampling strategy of environmental variables. The work done included both theoretical work and experimentation done in an applied and realistic context, allowing to follow closely the validation of the theoretical framework.

The results of the field testing of the approach were discussed in the last chapter of this thesis, where the project MATRAC-ACP that funded my PhD is presented and allowed to test in a real application all the elements developed and presented in this thesis.

This work provides several new contributions among which the main ones are the following:

- *study of the mathematical framework needed to extend spatial analysis to unstructured grids, used to discretise the sampling domain.* Especially when the domain to be considered is characterised by a very complex shape, the use of unstructured grids permits a more precisely fitting of the boundaries of this volume/surface. Considering supports with different shapes and sizes calls for adjustment on the estimation process, so the change of support models have been studied. The integration of the change of support model into the adaptive sampling strategy optimised by uncertainty contributed to the development of an important tool for environmental monitoring, where domains of interest often have irregular boundaries or also when an adaptive resolution is required. The details of theoretical aspects on change of support models have been discussed in Chapter 2 where we provide an experiment to evaluate the scenario of having a domain discretised by elements with different sizes.

- *A new strategy for selecting a set of locations in a domain with a reasoned criterion.* The proposed adaptive sampling optimises the positions of samples using information about a measure of uncertainty of the estimate itself, and this new approach improves the results of the traditional estimation processes,

as shown in the previous chapters. One of the main features of this strategy is the exploitation of data which are collected step by step to give more informative results at each iteration and to refine the representation of the spatial phenomenon.

The proposal solution has been applied in the framework of spatial monitoring, and it has been shown that the approach could integrate further optimisation criteria such as, for example, the traveled distance or other features relevant to maximise the efficiency of the surveying campaign. These results were discussed in Chapter 4. A lower number of points to be sampled, which leads approximately to the same reliability of representation of considering more random data, is desirable when the time and cost of the campaign want to be reduced.

The framework proposed has a wide applicability: whenever it is possible to associate a measure of uncertainty with an estimate, it will be possible to exploit this uncertainty to add information in a targeted way.

The proposed methodology can be also extended to other environmental situations such as soil and air. With respect to the investigation of marine waters, some limits and criticalities may emerge in these situations. Regarding air survey, the hypothesis of stationarity of the random field is less reasonable due to the variations caused by the wind. The same problem caused by sea currents also characterises water surveys, even if in a context of marine waters in ports this effect is under control. A possible solution in these cases is to reduce the time window in which the acquisition is carried out in order to guarantee a stationarity of the random field for a certain limited period of time. Regarding a soil survey, it could happen that not all points of the domain are reachable for the acquisition of samples. For this the unreachable areas should be excluded from the geometric model.

- *Definition and implementation of a new algorithm to discretise a tetrahedron exploiting the pros of the quasi-random generator.* This contribution was born from the need to compute the block-to-block covariance for the computation of coefficients of change of support but it could be very useful every time a tetrahedron must be discretised with a set of points that has specific characteristics (i.e. a low discrepancy value). The thesis provides a study of comparison between several methods for discretisation concluding that the best way to generate this set of points inside the tetrahedron is using Sobol' sequences generator. Chapter 3 describes the new method to generate points inside a

tetrahedron using quasi-random sequences. This allows to avoid the waste of generating points inside the unit cube and then to keep only the ones that are inside the tetrahedron avoiding a waste of points and time. In this way all the generated points can be considered.

- *Experimentation and validation of each theoretical aspect addressed by means of implementation of C++ code.* All functionalities implemented will be part of a geostatistical software. This software has the important characteristic to be fast providing results without hours of processing. The computational speed was one of the main goal when this work started because to provide results in real time the optimisation of the algorithms was a crucial feature considering large geometric models, made up of thousands of cells, and large amounts of data. During the work of this thesis also a graphical interface (GUI) to use all these functions and to visualise the results has been created. In particular, this GUI has a very useful application in the context of environmental monitoring when the expert needs a visualisation tool during the sampling survey.

Although the proposal adaptive sampling optimised by uncertainty applied on unstructured grids has provided very encouraging results the methodology needs some precautions.

Even if results are required in real time, the size of the geometric model or the amount of data could increase the computation time. To solve the problem of having a large geometric model, three solutions can be considered;

- splitting the sampling domain in sub-areas
- using a wider interval of discretisation for the model
- using an adaptive resolution to represent areas with less interest with a more coarse resolution.

Other limitation arises when the amount of data is very high. A possible solution could be to aggregate some data reducing the total number, for example averaging their values and considering only one point. One of the functions of the geostatistical tool we created allows to aggregate samples if they are too close each other.

Future Works

We are working to improve the geostatistical software adding new functionalities beyond those described in this thesis. Several improvements can be developed in the future:

- including robust statistical methods for the estimation of the variogram.
- the integration of directional variograms also in the estimation process i.e. considering them in the kriging interpolation.
- considering the geodetic distance when the neighbors must be identified in the kriging procedure.
- to integrate the function of cost associated with the displacement directly in the selection procedure.
- implementing other new geostatistical methodologies, such as co-kriging to be able to consider more than one variable at a time.

Furthermore, the method for generating quasi-random points inside tetrahedra (Chapter 3) could be extended to deal also with other 3D geometric shapes and be able to manage any support.

Acknowledgment

This research is financed by the project "Monitoraggio Adattivo in Tempo reale con Automatizzazione del Campionamento - Aree Costiere Portuali - MATRAC-ACP", funded by the EU Interreg VA Italia Francia Marittimo 2014-2020 - Asse prioritario del Programma 2 - Protezione e valorizzazione delle risorse naturali e culturali e gestione dei rischi - Obiettivo specifico della Prioritá d'Investimento 6C2-Accrescere la protezione delle acque marine nei porti.

A special thanks to Michela Spagnuolo and Marino Vetuschi Zuccolini for the opportunity that I was granted to work in the research sector in a innovative and stimulating topic. They have always supported me with their great experience and extensive knowledge during these PhD years.

CNR support and collaboration are gratefully acknowledged, with a special mention to Michela Mortara, Daniela Cabiddu and Simone Pittaluga.

Appendix A

Low-Discrepancy sequences

A.1 Sobol' Sequences

Sobol's quasi-random sequences are designed to generate a set of points that is uniformly distributed over the unit hypercube. Sobol' sequences are an example of quasi-random low-discrepancy sequences and they were first introduced by the Russian mathematician Ilya M. Sobol in 1967 [Sobol, 1976].

In order to facilitate the explanation of the Sobol's algorithm we consider just one dimension ($d = 1$). The aim is to generate a sequence of values $X = \{x_1, x_2, \ldots; 0 < x_i < 1\}$, with low discrepancy over the unit interval [Bratley and Fox, 1988]. A set of direction numbers $v_1, v_2, \ldots$ is needed to this purpose. Each v_i is a binary fraction that can be written in either of two ways:

$$v_i = 0.v_{i1}v_{i2}v_{i3}\ldots$$

where v_{ij} is the jth bit following the binary point in the expansion of v_i; or alternatively,

$$v_i = \frac{m_i}{2^i}$$

where m_i is an odd integer, $0 < m_i < 2^i$. In particular, we see from the second representation that $v_{ij} = 0$ if $j > i$.

To obtain v_i, we begin by choosing a polynomial with coefficients chosen from $\{0, 1\}$, which is a primitive polynomial [Knuth, 2014]. Thus we might choose,

$$P_q = x^q + a_1x^{q-1} + a_2x^{q-2} + \cdots + a_{q-1}x + 1$$

where each a_i is 0 or 1 and P_q is a primitive polynomial of degree q (if P_q is primitive, then a_q, the constant term, is necessarily equal to 1). Provided P_q is primitive, the

choice of polynomial is otherwise arbitrary. The number of primitive polynomials of degree q is $\phi(2q-1)/q$, where ϕ is the Euler function, and they are widely tabulated [Peterson et al., 1972].

Once we have chosen a polynomial, we use its coefficients to define a recurrence for calculating v_i; thus,

$$v_i = a_1 v_{i-1} \oplus a_2 v_{i-2} \oplus \cdots \oplus a_{q-1} v_{i-q+1} \oplus v_{i-q} \oplus \frac{v_{i-q}}{2^q} \qquad i > q$$

where $\oplus$ denotes a bit-by-bit exclusive-or operation, and the last term is v_{i-q} shifted right q places. Equivalently, we may express the recurrence in terms of the m_i and calculate:

$$m_i = 2a_1 m_{i-1} \oplus 2^2 a_2 m_{i-2} \oplus \cdots \oplus 2^{q-1} a_{q-1} m_{i-q+1} \oplus 2^q m_{i-q} \oplus m_{i-q} \qquad i > q \quad \text{(A.1)}$$

Using a primitive polynomial of degree q, the values of $m_1, m_2, \ldots, m_q$ can be chosen freely provided that each m_i is odd and $m_i < 2^i$; subsequent values $m_{q+1}, m_{q+2}, \ldots$ are then determined by the recurrence (A.1).

Finally, to generate the sequence $\{x_1, x_2, \ldots\}$ we can use

$$x_n = b_1 v_1 \oplus b_2 v_2 \oplus \ldots$$

where $\ldots b_3 b_2 b_1$ is the binary representation of n. This is Sobol's original method. Antonov and Saleev [Antonov and Saleev, 1979] prove that taking

$$x_n = g_1 v_1 \oplus g_2 v_2 \oplus \ldots \qquad \text{(A.2)}$$

where $\ldots g_3 g_2 g_1$ is the Gray code representation of n does not affect the asymptotic discrepancy. The Gray code has two properties:

1. The Gray code for n is obtained from the binary representation of n using

$$\ldots g_3 g_2 g_1 = \ldots b_3 b_2 b_1 \oplus \ldots b_4 b_3 b_2.$$

2. The Gray code for n and the Gray code for $n+1$ differ in only one position. If b, is the rightmost zero-bit in the binary representation of n (add a leading zero to n if there are no others), then g is the bit whose value changes.

Using these properties, and defining x_n by (A.2), we can calculate x_{n+1} in terms of x_n as

$$x_{n+1} = x_n \oplus v_c \qquad \text{(A.3)}$$

where b_c is the rightmost zero-bit in the binary representation of n. The Antonov-Saleev method is thus much faster than Sobol's original scheme. To start the recurrence, we take $x_0 = 0$.

Generalizing this procedure to d dimensions, we wish to generate sequences of quasi-random vectors $x_n^{(1)}, x_n^{(2)} \ldots, x_n^{(d)}$.

Sobol' [Sobol, 1976] proves that, to get $O(log^d N)$ discrepancy, it suffices to choose any d different primitive polynomials, to calculate d different sets of direction numbers as explained above, and then to generate each component $X_{(n)}^{(j)}$ of the quasi-random vector separately using the corresponding set of direction numbers.

A.2 R_d Sequence: Golden Ratio Sequence

Recurrence R-sequence falls into the category of quasi-random sequences that use irrational numbers to construct their basis (hyper)-parameters. Often, it is called Kronecker, Weyl or Richtmyer sequence [Press et al., 1996].

The canonical Kronecker recurrence sequence is defined as:

$$R_1(\alpha) : t_n = \{s_0 + n\alpha\}, \qquad n = 1, 2, 3, \ldots$$

where α is any irrational number. Note that the notation $\{y\}$ indicates the fractional part of y.

For $s_0 = 0$, the first few terms of the sequence $R_1(g)$, are:

$$t_n = 0.618,\ 0.234,\ 0.854,\ 0.472,\ 0.090,\ 0.708,\ 0.327,\ 0.944,\ 0.562,\ 0.180$$

It is important to note that the value of s_0 does not affect the overall characteristics of the sequence, and in nearly all cases is set to zero. The value of α that gives the lowest possible discrepancy is achieved if $\alpha = 1/g$, where g is the Golden Ratio. That is

$$g = \frac{\sqrt{5}+1}{2} \simeq 1.61803398875\ldots$$

The R_1 sequence, which is the Kronecker sequence using the Golden Ratio, is one of the best choices for one-dimensional quasi-random Monte Carlo integration methods.

There are many possible ways to generalise the Golden Ratio sequence to the $d-$dimensional space. For $d = 1$, $g_1 = 1.6180\ldots$ which is the canonical Golden Ratio; for $d = 2$, $g_2 = 1.3247\ldots$; and for $d = 3$, $g_3 = 1.2207\ldots$. The following parameter-free $d-$dimensional open (infinite) sequence $R_d(g)$ has excellent low

discrepancy characteristics:

$$t_n = \{n\alpha\}, \qquad n = 1, 2, 3, \ldots$$

$$\textit{where} \quad \alpha = (\frac{1}{g_d}, \frac{1}{g_d^2}, \frac{1}{g_d^3}, \ldots, \frac{1}{g_d^d})$$

and g_d *is the unique positive root of* $x^{d+1} = x + 1$

Appendix B

Pseudocode

B.1 Pseudocode of the Algorithms

Algorithm 1 Whole Adaptive Sampling Strategy (Section 4.2)

```
1: procedure INPUT PARAMETERS
2:     See Algorithm 2
3: procedure INITIALIZATION
4:     See Algorithm 3
5: procedure PRE-PROCESSING OF DATA
6:     procedure Sincronization
7:         See Algorithm 4
8:     procedure Transformation
9:         See Algorithm 7
10: procedure MAIN
11:     procedure Geostatistical analysis
12:         See Algorithm 6
13:     procedure Change of support model
14:         See Algorithm 11
```

Algorithm 2 INPUT PARAMETERS

1: $model \leftarrow$ 3D model of the area of interest (structured or unstructured grids)
2: $V_{master} \leftarrow (id_{sensore}, id_{Measure}) \leftarrow$ Select the MASTER variable
3: $N_{init} \leftarrow$ number of initial set of points
4: $N_{sim} \leftarrow$ number of simulations in SGS algorithm (Section 1.4.2)
5: $\delta \leftarrow$ frequency of sincronization ▷ to avoid plenty of data
6: $step$, $percent \leftarrow$ variogram's parameters
7: Type of optimization criterion: $Min\_U \leftarrow TRUE$ or $Min\_U\_Dist \leftarrow TRUE$ or $Min\_U\_Zone \leftarrow TRUE$
8: **if** $(Min\_U\_Dist \leftarrow TRUE)$ **then**
9: $N_{minDist} \leftarrow$ number of tets to select for the procedure of optimization
10: **if** $(Min\_U\_Zone \leftarrow TRUE)$ **then**
11: $N_{minZone} \leftarrow$ number of tets to select for the procedure of optimization
12: $radius \leftarrow$ to define the size of the sphere around the most uncertainty values
13: Type of discretization method for block-to-block variance: $Sobol \leftarrow TRUE$ or $GR \leftarrow TRUE$ or $regular \leftarrow TRUE$ or $CVT \leftarrow TRUE$
14: $N_disc \leftarrow$ number of discretization points
15: $stop \leftarrow$ parameter for the STOP criteria

Algorithm 3 INITIALIZATION: initial set of samples selected randomly to start the iterative procedure (Section 5.4.1).

1: $L = \{l_1, \ldots, l_{N_{init}}\} = \{(x_1, y_1, z_1), \ldots, (x_{N_{init}}, y_{N_{init}}, z_{N_{init}})\}$ ▷ Random coordinates of the initial set of locations
2: $Cp \leftarrow$current position (GPS information)
3: **while** (L is not empty) **do**
4: $l_i \leftarrow$ Select in L the position closer to Cp
5: ROV moves towards l_i
6: $V(l_i) \leftarrow$ sample environmental variable Z at position l_i
7: $Cp \leftarrow l_i$ ▷ Update the current position
8: $L \leftarrow L - l_i$
9: **return** $\{V(l_i), i = 1, \ldots, N_{init}\}$

Algorithm 4 SINCRONIZATION: at each waypoint the new sampled data are synchronized (Section 5.3).

1: $in\_data \leftarrow (tt, id_{sens}, id_{Measure}, M_{sens})$ ▷ Input Data Structure
2: $M_{master} \leftarrow M_{sens}$; measures of the master variable V_{master}
3: $tt_{master} \leftarrow tt$; timestamp relative to M_{sens} of the master variable V_{master}
4: $(X_{gps}, Y_{gps}, Z_{gps}) \leftarrow M_{sens}$; measures relative to GPS
5: $tt_{gps} \leftarrow tt$; timestamp relative to GPS
6: **for** (i in $1 : length(tt_{gps})$) **do**
7: **if** $tt_{gps}[i]$ is between any two values of tt_{master} **then**
8: $A \leftarrow tt_{master}[before]$
9: $B \leftarrow tt_{master}[after]$
10: $m_A \leftarrow M_{master}[before]$
11: $m_B \leftarrow M_{master}[after]$
12: $IM_{master}[i] \leftarrow LinearInterpol(m_A, m_B, A, B, tt_{gps}[i])$ ▷ See Algorithm 5
13: **else**
14: $IM_{master}[i] \leftarrow NULL$
15: $tt_{freq} \leftarrow seq\Big(from = tt_{gps}[1], to = tt_{gps}[length(tt_{gps})], by = \delta\Big)$
16: Select only the measures corresponding to tt_{freq}
17: **return** $out\_data \leftarrow (tt_{freq}, X_{gps}, Y_{gps}, Z_{gps}, IM_{master})$ ▷ Output Data Structure

Algorithm 5 Linear interpolation ($LinearInterpol$)

1: $measure_1$: first value of the measure
2: $measure_2$: second value of the measure
3: $interval_1$: first value of time interval related to $measure_1$
4: $interval_2$: second value of time interval related to $measure_2$
5: val : value of interval at each we want calculate a measure
6: $output \leftarrow measure_1 + (\frac{measure_2 - measure_1}{interval_2 - interval_1}) * (val - interval_1))$
return $output$

Algorithm 6 MAIN (Chapter 5)

$data \leftarrow (X_{gps}, Y_{gps}, Z_{gps}, IM_{master})$ ▷ interpolated values from *SINCRONIZATION* procedure (Algorithm 4)
$count \leftarrow 0$
$run \leftarrow 0$
while (stop==FALSE) **do**
    Compute and model the VARIOGRAM ▷ see Algorithm 8 and 9
    $SIM \leftarrow$ Sequential Gaussian Simulation (SGS), ▷ see Algorithm 10
    $E \leftarrow mean(SIM[, 1 : N_{sim}])$
    $U \leftarrow var(SIM[, 1 : N_{sim}])$
    $E_{bt} \leftarrow$ Back-Tranformation of E ▷ See Algorithm 13
    Draw maps of E_{bt} and U
    $mean\_U[run] \leftarrow mean(U)$
    $count \leftarrow count + 1 * (|mean\_U[run - 1] - mean\_U[run]| \leq threshold)$
    $stop \leftarrow$ Algorithm 17 using *count*
    if $(stop == FALSE)$ **then**
        Selection of next point to be sample:
        if $(Min\_U == TRUE)$ **then**
            $next \leftarrow$ See Algorithm 14
        if $(Min\_U\_Dist == TRUE)$ **then**
            $next \leftarrow$ See Algorithm 15
        if $(Min\_U\_Zone == TRUE)$ **then**
            $next \leftarrow$ See Algorithm 16
        Comunicate *next* to the ROV
        Vehicle reaches *next* and samples new data (also during the path)
        $data \leftarrow (X_{gps}, Y_{gps}, Z_{gps}, IM_{master})$ ▷ from *SINCRONIZATION* (Algorithm 4)
return (*data* of whole campaign, final E_{bt}, final U)

Algorithm 7 Normal Score Transformation (NST) (Section 1.4.1)

$m \leftarrow IM_{master}$ ▷ Input data
$NIM_{master} \leftarrow qqnorm(m)$ ▷ [R Core Team, 2020]
return NIM_{master}

Algorithm 8 Experimental Variogram with variable lag (Section 1.5.1)

$data \leftarrow (X_{gps}, Y_{gps}, Z_{gps}, NIM_{master})$ ▷ Normal Data from Algorithm 7
$h_0 \leftarrow 0$
for (k in $1:15$) **do**
    $\delta_k \leftarrow \delta_{k-1} + percent$ ▷ *percent* is a user's parameter
    $h_k \leftarrow h_{k-1} + \delta_k + \delta_{k-1}$
for (i in $1:length(data)$) **do**
    $\vec{x}_i \leftarrow (X_{gps}, Y_{gps}, Z_{gps})_i$
    for (j in $1:length(data)$) **do**
        $\vec{x}_j \leftarrow (X_{gps}, Y_{gps}, Z_{gps})_j$
        $dist \leftarrow ||\vec{x}_i - \vec{x}_j||$
        if $dist \in (\vec{h} - \vec{\delta}, \vec{h} + \vec{\delta})$ **then**
            $2\hat{\gamma}(\vec{h}) = VAR\Big[NIM_{master}(\vec{x}_i) - NIM_{master}(\vec{x}_j)\Big]$ ▷ Experimental Variogram
return $\hat{\gamma}(\vec{h})$

Algorithm 9 Fitting of the Experimental Variogram. (Section 1.5.1)

$\hat{\gamma}(\vec{h})$ from Algorithm 8
$nugget \leftarrow \hat{\gamma}(0)$
$sill \leftarrow 1 - nugget$
$range_{max} \leftarrow MAX(dist)$ ▷ Maximum distance
for $model \in \{$"Sph", "Gau", "Exp"$\}$ **do**
    for $range \in (0, range_{max})$ by $step$ **do** ▷ *step* is a user's parameter
        $T\_\gamma(h) \leftarrow$ Compute theoretical function with such $model$, $sill$, $nugget$ and $range$
        $MSE \leftarrow \sum_h \Big(T\_\gamma(h) - \hat{\gamma}(h)\Big)^2$
Select the $model$, $sill$, $nugget$ and $range$ with minimum MSE:
$Vario \leftarrow (model, sill, nugget, range)$
return $Vario$

Algorithm 10 Sequential Gaussian Simulation (Section 1.4.2). Each column of the output matrix corresponds to the estimates of the master variable in one realization of the simulations.

if (IM_{master} not follow Normal distribution) **then**
  $NormalData \leftarrow (X_{gps}, Y_{gps}, Z_{gps}, NIM_{master})$ ▷ See Algorithm 7
$Vario \leftarrow$ from Algorithm 9
$model \leftarrow$ 3D survey area (Unstructured grid)
$N_{tet} \leftarrow$ number of tetrahedrons in $model$
for (i in $1 : N_{sim}$) **do**
  $L_i \leftarrow (l_1^{(i)}, \ldots, l_{N_{tet}}^{(i)})$ random order of the tetrahedrons that will be visited (coordinates of the centroids of tetrahedrons).
  for (j in $1 : N_{tet}$) **do**
    Use kriging to obtain $\hat{V}(l_j^{(i)})$ and $\hat{\sigma}_K^2(l_j^{(i)})$
    Draw a value $v_j^{(i)}$ at random from a normal distribution with mean $\hat{V}(l_j^{(i)})$ and variance $\hat{\sigma}_K^2(l_j^{(i)})$
    Point $l_j^{(i)}$ and value $v_j^{(i)}$ are added to the SGS set of known values.
return $SIM \leftarrow$ matrix of (N_{tet} x N_{sim}) elements

Algorithm 11 Computation of change of support coefficients with DGM (Section 2.2.2).

$N_{disc} \leftarrow$ Number of points to discretize a tetrahedron ▷ User's parameter
$P = 10$
$\{\phi_p, p = 0, \ldots, P\} \leftarrow$ Hermite Coefficients
for (i in $1 : N_{tet}$) **do**
  $\{p_1, \ldots, p_{N_{disc}}\}_i \leftarrow$ discretization points of tetrahedron i using for example Sobol's algorithm or GR
  Compute the distances among $p_k, k = 1, \ldots, N_{disc}$
  Compute covariance model from original data
  $\sum_{p=0}^{P} (\phi_p r_i^p)^2 = \frac{1}{|v_i|^2} \int_{v_i} \int_{v_i} C(x - x^{'}) dx dx^{'}$
  $r_i \leftarrow$ Use L-BFGS algorithm [Liu and Nocedal, 1989]
return $\{r_i, i = 1, \ldots, N_{tet}\}$

Algorithm 12 Computation of change of support coefficients with DGM-2 (Section 2.2.3).

$N_{disc} \leftarrow$ Number of points to discretize a tetrahedron ▷ User's parameter
for (i in $1 : N_{tet}$) **do**
  $\{p_1, \ldots, p_{N_{disc}}\}_i \leftarrow$ discretization points of tetrahedron i using for example Sobol's algorithm or GR
  Compute the distances among $p_k, k = 1, \ldots, N_{disc}$
  Select from the fitted variogram, $Vario$, the corresponding value of $\gamma(h)$ for each distance
  $VAR_{b2b} \leftarrow$ Average the $\gamma(h)$ of all pairs of the discretization points
  $r_i \leftarrow \sqrt{VAR_{b2b}}$ ▷ Change of Support Coefficients
return $\{r_i, i = 1, \ldots, N_{tet}\}$

Algorithm 13 Back-Tranformation with Hermite (Section 2.13).

$data \leftarrow (X_{gps}, Y_{gps}, Z_{gps}, IM_{master})$
$E \leftarrow$ Estimation Map
$r_i \leftarrow$ Algorithm 11 or 12
$P = 10$
$\{\phi_p, p = 0, \ldots, P\} \leftarrow$ Hermite Coefficients
$\{H_p(Y), p = 0, \ldots, P\} \leftarrow$ Hermite Polynomials
for (i in $1 : N_{tet}$) **do**
    $E_{bt}[i] \leftarrow \sum_{p=0}^{P} \phi_p r_i^p H_p(E[i])$
return E_{bt}

Algorithm 14 Selection of the next point optimizing only the uncertainty

$U \leftarrow$ Uncertainty Map
$next \leftarrow$ centroid of the tetrahedron with $max(U)$
return $next$

Algorithm 15 Selection of the next point optimizing uncertainty and displacement (Section 4.2.2).

$U \leftarrow$ Uncertainty Map
$Cp \leftarrow$ current position
for (i in $1 : N_{tet}$) **do**
    $distance = ||C_i - Cp||$ ▷ C_i is the centroid of the i-th tetrahedron
$sorted \leftarrow$ sort tetrahedra by uncertainty U
$(t_1, \ldots, t_{N_{minDist}}) \leftarrow$ the first $N_{minDist}$ tetrahedra in $sorted$
$next \leftarrow$ Select the tetrahedron with minimum $distance$ among $(t_1, \ldots, t_{N_{minDist}})$
return $next$

Algorithm 16 Selection of the next point as the center of the sphere with the highest uncertainty (Section 4.2.2).

$U \leftarrow$ Uncertainty Map
$sorted \leftarrow$ sort tets by uncertainty U
$(t_1, \ldots, t_{N_{minZone}}) \leftarrow$ the first $N_{minZone}$ tetrahedra in $sorted$
for (i in $1 : N_{minZone}$) **do**
    $U_i^{(zone)} \leftarrow$ mean-uncertainty of the sphere ($radius$) centered in t_i ▷ It is the mean of uncertainties of each tetrahedron belonging to the sphere
$next \leftarrow$ the center of the sphere with maximum $U_i^{(zone)}$
return $next$

Algorithm 17 STOP criterion (Section 5.4.3).

```
1: stop ← user's parameter
2: if (count == stop) then
3:     Warning message to the expert
4:     if Manual stop is selected by expert then
5:         stop ← TRUE
6:     else
7:         stop ← FALSE
8: else
9:     stop ← FALSE
       return stop
```

Appendix C

Plots of the Experiment on DGM's Effectiveness

	$\sigma = 1$		$\sigma = 50$	
	N=100	N=200	N=100	N=200
Model	Gaussian	Gaussian	Gaussian	Gaussian
Nugget	0.006	0.010	0.510	0.560
Sill	0.994	0.990	0.490	0.440
Range	60.378	59.191	74.239	70.959

Table C.1: Parameters of the fitted variograms

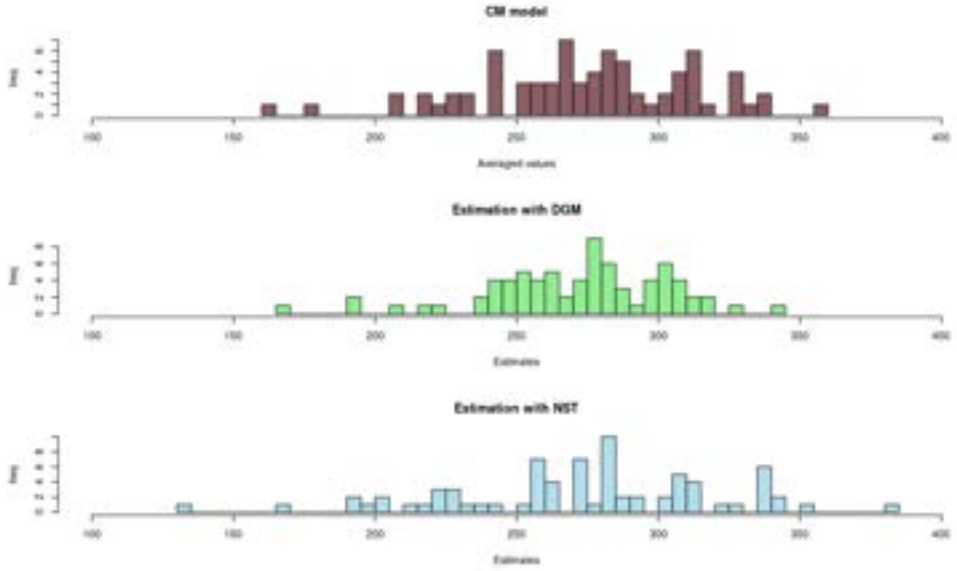


Figure C.1: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model. The number of sampled points is $N = 100$ and the function in (2.33) has as variance of ϵ equal to one ($\sigma = 1$).

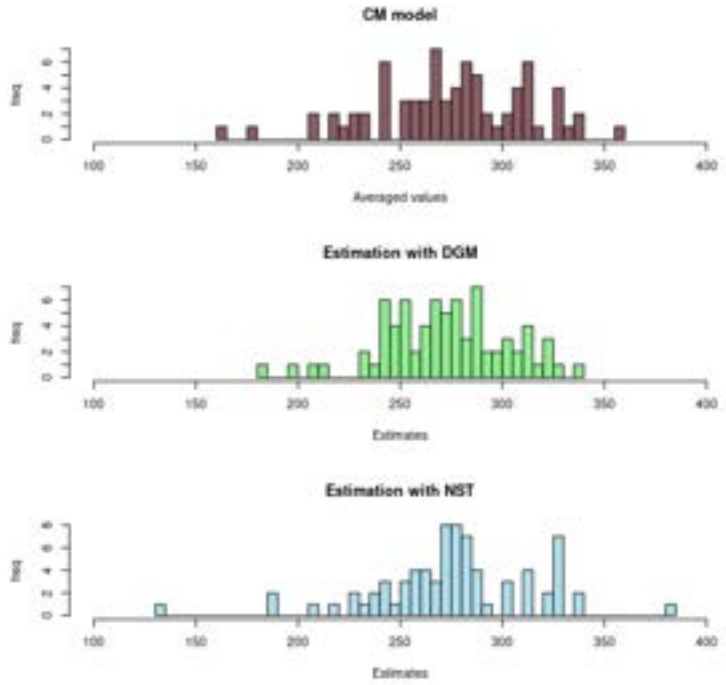


Figure C.2: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model. The number of sampled points is $N = 200$ and the function in (2.33) has as variance of ϵ equal to one ($\sigma = 1$).

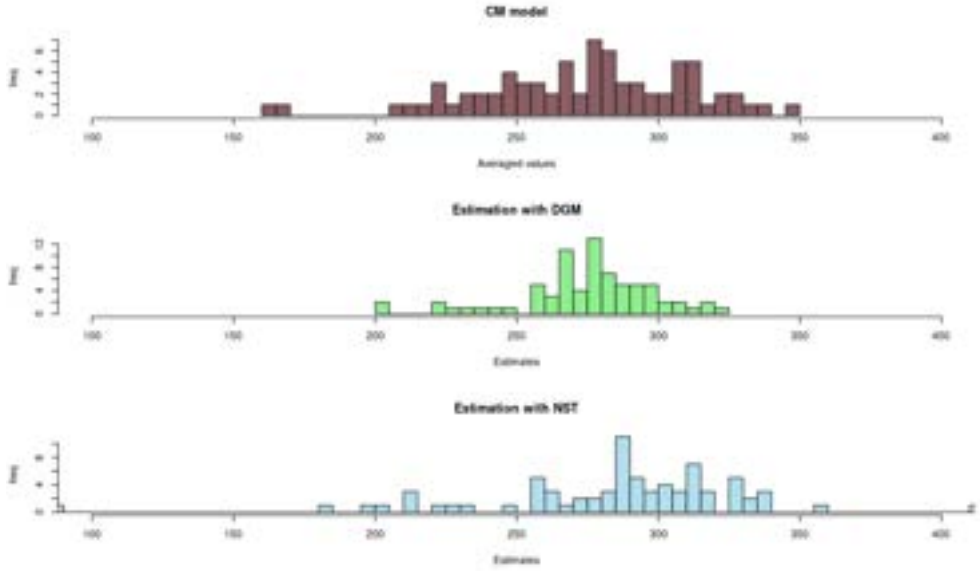


Figure C.3: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model. The number of sampled points is $N = 100$ and the function in (2.33) has as variance of ϵ equal to fifty ($\sigma = 50$).

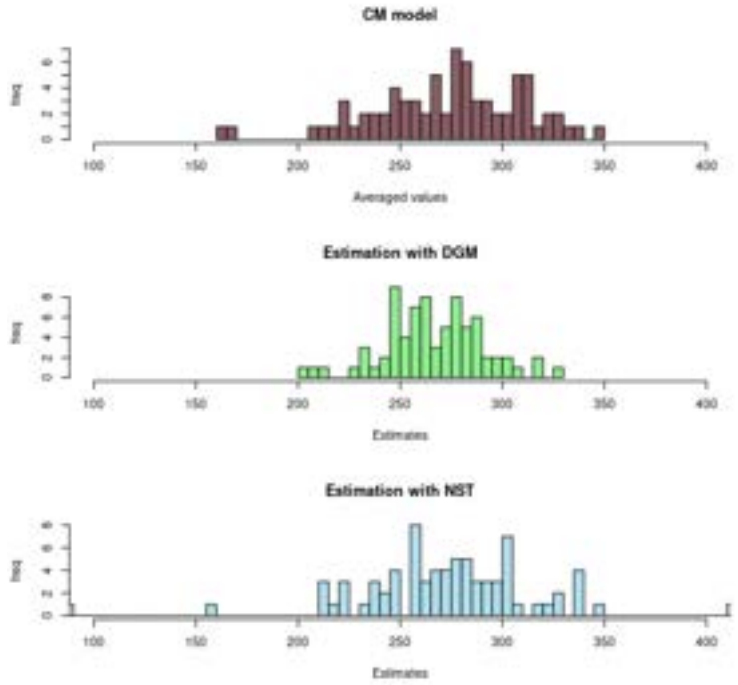


Figure C.4: Comparison between the results of the estimations using DGM and NST for the back transformation with the averaged synthetic field on the coarse model. The number of sampled points is $N = 200$ and the function in (2.33) has as variance of ϵ equal to fifty ($\sigma = 50$).

Bibliography

Alfsen, E. M. (2012). *Compact convex sets and boundary integrals*, volume 57. Springer Science & Business Media.

Antonov, I. A. and Saleev, V. (1979). An economic method of computing lpτ-sequences. *USSR Computational Mathematics and Mathematical Physics*, 19(1):252–256.

Banerjee, S. (2005). On geodetic distance computations in spatial modeling. *Biometrics*, 61(2):617–625.

Beck, J. (1994). Probabilistic diophantine approximation, i. kronecker sequences. *Annals of Mathematics*, pages 449–502.

Berg, M. d., Kreveld, M. v., Overmars, M., and Schwarzkopf, O. (1997). Computational geometry. In *Computational Geometry*, pages 1–17. Springer.

Berndt, A. E. (2020). Sampling methods. *Journal of Human Lactation*, 36(2):224–226.

Berretta, S., Cabiddu, D., Mortara, M., and Spagnuolo, M. (2020). Sea monitoring made simple and efficient. *ERCIM News*, 2020(123).

Berretta, S., Cabiddu, D., Pittaluga, S., Mortara, M., Spagnuolo, M., and Zuccolini, M. V. (2018a). Adaptive environmental sampling: The interplay between geostatistics and geometry. In *STAG*, pages 133–140.

Berretta, S., Cabiddu, D., Pittaluga, S., Mortara, M., Spagnuolo, M., and Zuccolini, M. V. (2018b). Adaptive sampling of enviromental variables (asev). *IMATI Report Series*, (18-06).

Bratley, P. and Fox, B. L. (1988). Algorithm 659: Implementing sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software (TOMS)*, 14(1):88–100.

Caccia, M., Ferretti, R., Odetti, A., Bruzzone, G., Spagnuolo, M., Mortara, M., Berretta, S., Cabiddu, D., Pittaluga, S., Zuccolini, M. V., et al. (2019). Robotics and adaptive sampling techniques for harbor waters monitoring: the matrac-acp project. In *OCEANS 2019-Marseille*, pages 1–8. IEEE.

Caers, J. (2011). *Modeling uncertainty in the earth sciences*. John Wiley & Sons.

Chilès, J. and Delfiner, P. (2012). *Geostatistics: Modeling Spatial Uncertainty,*, pages 705–14.

Chiles, J.-P. and Delfiner, P. (2009). *Geostatistics: modeling spatial uncertainty*, volume 497. John Wiley & Sons.

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

Crane, K., Livesu, M., Puppo, E., and Qin, Y. (2020). A survey of algorithms for geodesic paths and distances. *arXiv preprint arXiv:2007.10430*.

Cressie, N. (1985). Fitting variogram models by weighted least squares. *Journal of the International Association for Mathematical Geology*, 17(5):563–586.

Cressie, N. and Hawkins, D. M. (1980). Robust estimation of the variogram: I. *Journal of the international Association for Mathematical Geology*, 12(2):115–125.

DeGroot, M. H. and Schervish, M. J. (2012). *Probability and Statistics*. Pearson Education.

Deutsch, C. V., Journel, A. G., et al. (1992). Geostatistical software library and user's guide. *Oxford University Press*, 8(91):0–1.

Du, Q., Faber, V., and Gunzburger, M. (1999). Centroidal voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637–676.

Emery, X. (2007). On some consistency conditions for geostatistical change-of-support models. *Mathematical Geology*, 39(2):205–223.

Etikan, I. and Bala, K. (2017). Sampling and sampling methods. *Biometrics & Biostatistics International Journal*, 5(6):00149.

Faure, H. (1986). On the star-discrepancy of generalized hammersley sequences in two dimensions. *Monatshefte für Mathematik*, 101(4):291–300.

Fuhg, J. N., Fau, A., and Nackenhorst, U. (2021). State-of-the-art and comparative review of adaptive sampling methods for kriging. *Archives of Computational Methods in Engineering*, 28(4):2689–2747.

Gandar, B., Loosli, G., and Deffuant, G. (2010). Sample dispersion is better than sample discrepancy for classification.

Genton, M. G. (1998). Highly robust variogram estimation. *Mathematical Geology*, 30(2):213–221.

Goovaerts, P. (2016). *Sample Support*. In Encyclopedia of Environmetrics, John Wiley Sons, Ltd.

Gotway Crawford, C. A. and Young, L. (2005). Change of support: an interdisciplinary challenge. In *Geostatistics for Environmental Applications*, pages 1–13. Springer.

Grazzini, J., Soille, P., and Bielski, C. (2007). On the use of geodesic distances for spatial interpolation. In *Proceedings of the 9th International Conference on GeoComputation, GeoComputation, Maynooth, Ireland*. Citeseer.

Halton, J. H. (1960). On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90.

Harvey, J. B., Ryan, J. P., Marin III, R., Preston, C. M., Alvarado, N., Scholin, C. A., and Vrijenhoek, R. C. (2012). Robotic sampling, in situ monitoring and molecular detection of marine zooplankton. *Journal of Experimental Marine Biology and Ecology*, 413:60–70.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hyndman, R. J. (2011). Moving averages. *International Encyclopedia of Statistical Science*, pages 866—869.

Isaaks, E. H. and Srivastava, R. M. (1989). Applied geostatistics. *Oxford University Press*, 561.

Keith, L. H. (2017). *Environmental sampling and analysis: a practical guide*. Routledge.

Knuth, D. E. (2014). *Art of computer programming, volume 2: Seminumerical algorithms*. Addison-Wesley Professional.

Kothari, C. R. (2004). *Research methodology: Methods and techniques*. New Age International.

Lark, R. (2000). A comparison of some robust estimators of the variogram for use in soil survey. *European journal of soil science*, 51(1):137–157.

Levy, P. S. and Lemeshow, S. (2013). *Sampling of populations: methods and applications*. John Wiley & Sons.

Liu, D. C. and Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45(1):503–528.

Livesu, M. (2019). cinolib: a generic programming header only c++ library for processing polygonal and polyhedral meshes. In *Transactions on Computational Science XXXIV*, pages 64–76. Springer.

Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, 58(8):1246–1266.

Mitchell, J. S., Mount, D. M., and Papadimitriou, C. H. (1987). The discrete geodesic problem. *SIAM Journal on Computing*, 16(4):647–668.

Niederreiter, H. (1978). Quasi-monte carlo methods and pseudo-random numbers. *Bulletin of the American Mathematical Society*, 84(6):957–1041.

Nocedal, J. and Wright, S. J. (1999). *Numerical optimization*. Springer.

Nunes, R. and Almeida, J. A. (2010). Parallelization of sequential gaussian, indicator and direct simulation algorithms. *Computers & Geosciences*, 36(8):1042–1052.

Omre, H. (1984). The variogram and its estimation. In *Geostatistics for natural resources characterization*, pages 107–125. Springer.

Ortiz, J. M., Oz, B., and Deutsch, C. V. (2005). A step by step guide to bi-gaussian disjunctive kriging. In *Geostatistics Banff 2004*, pages 1097–1102. Springer.

Peterson, W. W., Peterson, W., Weldon, E. J., and Weldon, E. J. (1972). Error-correcting codes.

Peyré, G. and Cohen, L. D. (2009). Geodesic methods for shape and surface processing. *Advances in Computational Vision and Medical Image Processing*, pages 29–56.

Preparata, F. P. and Shamos, M. I. (2012). *Computational geometry: an introduction*. Springer Science & Business Media.

Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. (1996). *Numerical recipes in C*. Cambridge University Press.

Press, W. H. and Teukolsky, S. A. (1989). Quasi-(that is, sub-) random numbers. *Computers in Physics*, 3(6):76–79.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ramshaw, L. (1981). On the discrepancy of the sequence formed by the multiples of an irrational number. *Journal of Number Theory*, 13(2):138–175.

Richtmyer, R. D. (1951). The evaluation of definite integrals, and a quasi-monte-carlo method based on the properties of algebraic numbers. Technical report, Los Alamos Scientific Lab.

Rivoirard, J. (1994). *Introduction to disjunctive kriging and non-linear geostatistics*. Number 551.021 R626i. Clarendon Press,.

Roberts, M. (2019). Evenly distributing points in a triangle. `http://extremelearning.com.au/evenly-distributing-points-in-a-triangle/`.

Rocchini, C. and Cignoni, P. (2000). Generating random points in a tetrahedron. *Journal of graphics Tools*, 5(4):9–12.

Rocchini, D., Hortal, J., Lengyel, S., Lobo, J. M., Jimenez-Valverde, A., Ricotta, C., Bacaro, G., and Chiarucci, A. (2011). Accounting for uncertainty when mapping species distributions: the need for maps of ignorance. *Progress in Physical Geography*, 35(2):211–226.

Si, H. (2015). Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software (TOMS)*, 41(2):1–36.

Singh, A. S. and Masuku, M. B. (2014). Sampling techniques & determination of sample size in applied statistics research: An overview. *International Journal of Economics, Commerce and Management*, 2(11):1–22.

Sobol, I. M. (1976). Uniformly distributed sequences with an additional uniform property. *USSR Computational Mathematics and Mathematical Physics*, 16(5):236–242.

Taherdoost, H. (2016). Sampling methods in research methodology; how to choose a sampling technique for research. *How to choose a sampling technique for research (April 10, 2016)*.

Turk, G. (1990). Generation random points in triangles. *Graphic Gems*.

Van Der Corput, J. (1935). Verteilungsfunktionen i & ii. In *Nederl. Akad. Wetensch. Proc.*, volume 38, pages 1058–1066.

Webster, R. and Oliver, M. A. (2007). *Geostatistics for environmental scientists*. John Wiley & Sons.

Yamahara, K. M., Preston, C. M., Birch, J., Walz, K., Marin III, R., Jensen, S., Pargett, D., Roman, B., Ussler III, W., Zhang, Y., et al. (2019). In situ autonomous acquisition and preservation of marine environmental dna using an autonomous underwater vehicle. *Frontiers in Marine Science*, 6:373.

Zaytsev, V. (2016). *Méthodes stochastiques pour la modélisation d'incertitudes sur les maillages non structurés*. PhD thesis, Paris Sciences et Lettres (ComUE).

Zaytsev, V., Biver, P., Wackernagel, H., and Allard, D. (2016). Change-of-support models on irregular grids for geostatistical simulation. *Mathematical Geosciences*, 48(4):353–369.

Zhang, C. (2007). *Fundamentals of environmental sampling and analysis*. John Wiley & Sons.