INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI film
the text directly from the original or copy submitted. Thus, some thesis and
dissertation copies are in typewriter face, while others may be from any type of
computer printer.

The quality of this reproduction is dependent upon the quality of the
copy submitted. Broken or indistinct print, colored or poor quality illustrations
and photographs, print bleedthrough, substandard margins, and improper
alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript
and there are missing pages, these will be noted. Also, if unauthorized
copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by
sectioning the original, beginning at the upper left-hand corner and continuing
from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced
xerographically in this copy. Higher quality 6" x 9" black and white
photographic prints are available for any photographs or illustrations appearing
in this copy for an additional charge. Contact UMI directly to order.
NEW METHODS FOR SPATIAL STATISTICS IN
GEOGRAPHIC INFORMATION SYSTEMS

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in the Department of Civil and Environmental Engineering and Geodetic Science of The Ohio State University

By

Yaron A. Felus, B.S., M.S.

* * * * *

The Ohio State University

2001

Dissertation Committee:

Prof. Alan Saalfeld, Adviser
Prof. Burkhard Schaffrin, Co-Adviser
Prof. Noel Cressie
Prof. Beáta Csathó

Approved by:

Prof. Alan Saalfeld
Adviser
Graduate program in Geodetic Science and Surveying
ABSTRACT

This thesis is concerned with the development of new methods to implement advanced spatial statistics procedures within Geographic Information Systems. Two approaches are investigated; the first uses Delaunay triangulation data structure to select an appropriate subset of the data and perform Ordinary Kriging interpolation. The second studies an advanced spatial statistics procedure, the optimal biased kriging, which is more efficient in terms of the Mean Squared Prediction Error.

Spatial statistical prediction, also called Kriging, has been proved to be an accurate and reliable interpolation method. However, geostatistical interpolation is a computationally very consuming process since it involves the inversion of a $n \times n$ matrix where $n$ is the number of sampled points. The common way to overcome this problem is to include only "nearby" points in the Kriging process. Nevertheless, the choice of the Kriging points, we will call it support, has a significant effect on the accuracy of our prediction. Consequently, we will investigate the use of a well-established computational geometry algorithm - the Delaunay triangulation - as a data structure to select our interpolation support. During this investigation we will develop an efficient algorithm and build up understanding about the statistical effect of a limited neighborhood on the interpolation. Moreover, we will test and evaluate the proposed innovative method using newly acquired aeromagnetic data collected at the West Antarctic mountains.

Optimal Biased Kriging is an efficient geostatistical method that gives up
unbiasedness to gain improvement in the mean squared prediction error. We will apply this theory on a set of laser-scanning topographic data. In the implementation we will use a relatively new spatial coherency measure, the homeogram, also known as the non-centered covariance function. Moreover, we use a pre-interpolation spatial sorting to obtain a band-limited sparse coherency matrix. The sparseness of the coherency matrix is used to enhance the interpolation algorithm.
ACKNOWLEDGMENT

I wish to express my deepest appreciation to my advisors: Professor Alan Saalfeld for introducing me to the topics of geocomputation and for his extraordinary patience and support throughout the course of my studies. Professor Burkhard Schaffrin for his invaluable input, ideas, and fruitful discussions that solidified much of the material in this dissertation; his scholarly and constructive remarks really helped in shaping this work. I am also deeply indebted to my committee members; Professor Noel Cressie for his tips and thoughtful comments that directed me during these research studies. I owe much gratitude to the discussions I had with Professor Beata Csatho whose views are always enlightening and provided a strong impetus for this research. My deepest thanks to Professor Terry Wilson who supported me throughout the research and gave me a lifetime experience with the research expedition in Antarctica. I thank the friends, students, colleagues at the Geomatics Laboratory for Ice Dynamics (GLID) and at the Byrd polar research center for providing environment inductive to learning and high-quality research. I am also grateful to many people at the Ohio-State University: Professor Toni Schenk for the constructive conversations and suggestions, Professor Ron Li for his support and guidance, Professor Ralph Von-Frese for the stimulating discussion and advice, and Irene Tesfai for being available always and offering a warm advice. Finally, I will be ever grateful for the love of my parents, my children, and especially my spouse. Without you all being there, this dissertation would never have been completed.
VITA

June 26, 1968............. Born in Holon, Israel
1986-1990.............. B.S. in Electrical and Computer Engineering at the Ben-Gurion University of the Negev, Beer-Sheva, Israel
1994-1995.............. M.S. at the Geoinformatics Department of the International Institute for Aerospace Survey and Earth Sciences (ITC), Enschede, The Netherlands
1996-1998.............. Diploma studies for Licensed Surveyor Certificate In the Department of Geodetic Science and Civil Engineering, Technion, Haifa, Israel
1996-1998.............. Research and Development manager at Halperin-Felus Surveying and Photogrammetry LTD, Israel
1998-2001.............. Graduate Research Associate at the Byrd Polar Research Center of the Ohio State University, Columbus, Ohio
2000-2001.............. Graduate Teaching Assistant at the Department of Civil and Environmental Engineering and Geodetic Science of the Ohio State University, Columbus, Ohio

FIELDS OF STUDY

Major Field: Geodetic Science and Surveying
TABLE OF CONTENTS

ABSTRACT......................................................................................................................ii
ACKNOWLEDGMENT ..................................................................................................iv
VITA.................................................................................................................................iv
TABLE OF CONTENTS....................................................................................................v
LIST OF FIGURES.........................................................................................................vii
LIST OF TABLES.............................................................................................................x
LIST OF ACRONYMS AND NOTATION......................................................................xi

1. INTRODUCTION........................................................................................................... 1
  1.1 Spatial statistics and GIS .......................................................................................... 1
  1.2 Research problems ................................................................................................... 4
  1.3 Description of our case studies ................................................................................. 7
    1.3.1 TAMARA - Aeromagnetic data......................................................................... 7
    1.3.2 Ocean city- Laser scanning data ...................................................................... 11
  1.4 Organization of the report....................................................................................... 13

2. GEOSTATISTICAL INTERPOLATION METHODS ......................................................... 15
  2.1 Review of interpolation methods ............................................................................ 15
    2.1.1 Classification of methods................................................................................. 15
    2.1.2 Inverse distance weighting ............................................................................... 17
    2.1.3 Trend surface analysis ...................................................................................... 18
    2.1.4 Minimum curvature interpolation .................................................................... 19
    2.1.5 Area stealing interpolation .............................................................................. 21
    2.1.6 The geostatistical paradigm (Kriging) ............................................................. 23
  2.2 Estimation of spatial coherency functions .............................................................. 25
    2.2.1 Introduction and basic stochastic assumptions ................................................ 25
    2.2.2 Estimation of spatial coherency functions ....................................................... 28
    2.2.3 Variogram Vs. Covariance function Vs. Homeogram..................................... 31
    2.2.4 Fitting a model to the spatial coherency functions .......................................... 32
    2.2.5 Experiments with fitting spatial coherency functions ...................................... 37
  2.3 The geostatistical paradigm ................................................................................... 47
3. SPATIAL DATA STRUCTURE IN SUPPORT FOR GEOSTATISTICAL ALGORITHMS

3.1 Search neighborhood of geostatistical algorithms .................................................. 51
  3.1.1 Review of current search strategies ................................................................. 53
3.2 Delaunay triangulation data structure ............................................................... 56
  3.2.1 Introduction to triangulation ............................................................................ 56
  3.2.2 Properties of the Delaunay triangulation: ......................................................... 57
  3.2.3 Using TIN for piecewise linear interpolation ................................................. 61
  3.2.4 Defining the order of Delaunay neighborhood ................................................ 64
3.3 Voronoi diagram ..................................................................................................... 69
  3.3.1 Properties of the Voronoi diagram................................................................... 70
3.4 Computational complexity of neighborhood search algorithms ............................. 72
  3.4.1 The complexity of computing the Voronoi diagram and Delaunay triangulation ................................................................................................................................... 74
  3.4.2 Complexity of points in a planar subdivision algorithm .................................... 78
  3.4.3 Complexity of the nearest neighbor search algorithm ..................................... 80
  3.4.4 Comparing the complexity search methods, with a note on Hessami et al. (2001) ........................................................................................................................ 83
3.5 Applying the Delaunay structure as a search method: Recursive algorithm ......... 85

4. ON THE NEIGHBORHOOD OF KRIGING INTERPOLATION .................................. 87

  4.1 Introduction ............................................................................................................. 87
  4.1.1 Neighborhood and Markov Random Field ...................................................... 87
  4.1.2 Finite domain Kriging ...................................................................................... 90
  4.2 Parameters for neighborhood selection .............................................................. 96
  4.3 Data analysis using Fourier theory ....................................................................... 103
  4.3.1 Detrending the data, minimum point set required ........................................... 104
  4.3.2 Example: Applying the sampling theory to the magnetic data ...................... 106
  4.3.3 Neighborhood selection in the Fourier domain ............................................ 109
  4.4 Experiments performed on the TAMARA case study dataset ............................. 111

5. EXPERIMENTS WITH OPTIMAL BIASED KRIGING .............................................. 116

  5.1 Concept and formulas .......................................................................................... 116
  5.2 Spatial coherency matrix inversion ..................................................................... 124
  5.2.1 Stability of the coherency matrix .................................................................... 124
  5.2.1 Matrix inversion methods ............................................................................... 126
5.2.3 Constructing a band-limited structured matrix ............................................ 128
5.3 Implementation and experimental results ........................................................... 132

6. CONCLUSIONS AND FUTURE RESEARCH ..................................................... 141

APPENDIX A, DATA SETS AND PROGRAM DESCRIPTIONS ................................. 146

A.1 Data sets and their descriptions ........................................................................ 146
A.1.1 Magnetic data file: ...................................................................................... 146
A.1.2 Laser scanning data set: .............................................................................. 147
A.1.3 Ancillary data sets: ..................................................................................... 148
A.2 Matlab programs .............................................................................................. 148
A.2.1 Programs used in Chapter 2: ....................................................................... 148
A.2.2 Programs used in Chapter 3: ....................................................................... 149
A.2.3 Programs used in Chapter 4: ....................................................................... 149
A.2.4 Programs used in Chapter 5: ....................................................................... 150

APPENDIX B, Mathematical proofs ........................................................................ 151

B.1 Ordinary Kriging prediction ............................................................................. 151
B.2 Optimal Biased kriging predictor (homBLIP) .................................................... 157

APPENDIX C, FIGURES OF COMPUTATIONAL GEOMETRY NEIGHBORHOOD
PROBLEM ............................................................................................................. 161

REFERENCES ......................................................................................................... 163
LIST OF FIGURES

Figure 1.1: Interaction of GIS and geostatistics................................................................. 2
Figure 1.2: Study area in West Antarctica, ................................................................. 9
Figure 1.3: Laser data collected on Assateague Island, Maryland. .......................... 13
Figure 2.1: The area stealing interpolation.................................................................... 22
Figure 2.2: Examples for semi-variogram and covariance function/homeogram......... 35
Figure 2.3: Coherency functions of the test magnetic.................................................... 39
Figure 2.4: Coherency functions of the test magnetic (limited distance)...................... 39
Figure 2.5: Trend analysis of the magnetic data............................................................. 40
Figure 2.6: Coherency functions of the residuals of the detrended magnetic data..... 40
Figure 2.7: Coherency functions of the laser scanning test data.................................... 41
Figure 2.8: Coherency functions of the laser scanning test data after detrending.... 41
Figure 2.9: Comparison of the exponential, Gaussian, and prismato-magnetic models. 44
Figure 2.10: Model fitting to the laser scanning test data............................................. 46
Figure 3.1: Common search strategies.......................................................................... 56
Figure 3.2: In-circle of the Delaunay triangulation....................................................... 58
Figure 3.4: The ambiguity of the Delaunay triangulation............................................. 60
Figure 3.5: Counter-example for the case of the closest point to P not in the triangle... 61
Figure 3.6: Linear interpolation in a triangle adapted from Terei (1999)...................... 62
Figure 3.7: Orders of Delaunay neighborhood............................................................. 65
Figure 3.9: Voronoi diagram of a set of points.......................................................... 70
Figure 3.10: Voronoi diagram neighborhood may cause extrapolation inadvertently.... 72
Figure 3.11: Fortune's algorithm figure taken from de Berg (1997). ......................... 75
Figure 3.12: A circle event sequence adapted from de Berg (1997)... ........................ 76
Figure 3.13: Four points that lie on the same circle, and their Voronoi diagram.......... 78
Figure 3.14: Walking method to find the triangle which includes the query point....... 80
Figure 3.15: Spiral search of nearest neighbor using the cell Method

Figure 4.1: Kriging weights of as a function of the distance from the data string

Figure 4.2: Example to compare interpolation and extrapolation

Figure 4.3: Linear semi-variogram as defined in (4.4)

Figure 4.4: The computed MSPE as a function of X3 for the linear model at (4.4)

Figure 4.5: The computed MSPE as a function of the magnetic data

Figure 4.6: The effect of the extrapolation error in 1st order polynomial fitting

Figure 4.7: Our magnetic signal and its Fourier transform

Figure 4.8: Power spectrum of the magnetic data

Figure 4.9: Rect function and its frequency representation

Figure 4.10: Experiments with magnetic data

Figure 5.1: The covariance matrix and its associated graph

Figure 5.2: Peano Key or Morton Sorting algorithm

Figure 5.3: Peano Key / Morton Ordering, spatial sorting

Figure 5.4: Data points before and after applying Morton Ordering

Figure 5.5: Laser data points with the 3rd order polynomial surface
LIST OF TABLES

Table 1.1: Specification of the laser scanner data............................................................12
Table 2.1: Mathematical models for Variogram/covariance function..............................34
Table 2.2 Results of model fitting to the laser scanning test data.....................................45
Table 3.1: Geometric region of influence from Midtbø (1993)........................................66
Table 3.2: Properties of the equilateral triangle and square mesh..................................69
Table 3.3. Delaunay triangulation algorithms and the computational complexity............74
Table 3.5: Flow-chart of the developed interpolation process.........................................86
Table 4.1: Results of experiments on magnetic data with no detrending.......................113
Table 4.2: Results of experiments on magnetic data with detrending............................113
Table 5.1.1: original covariance matrix $K$.....................................................................134
Table 5.1.2: Covariance matrix $K$ computed from data after applying Morton ordering..........................................................134
Table 5.2: Results of the comparison between OBK and OK perditions methods large data set................................................................................................................136
Table 5.3: Results of the comparison between OBK and OK perditions methods small data set no. 1................................................................................................................139
Table 5.4: Results of the comparison between OBK and OK perditions methods small data set no. 2................................................................................................................139
LIST OF ACRONYMS AND NOTATION

Notations are sometimes confusing in geodetic science and especially when they are coupled with statistical and mathematical notations. I have tried to summarize most of the symbols that I am using in this dissertation, though there may be a few that I missed. Those are given in the relevant paragraph.

**Acronyms:**

BLUUE - Best Linear Uniformly Unbiased Estimation.

BIQUUE - Best Invariant Quadratic Uniformly Unbiased Estimation.

BLIP - Best Linear Prediction.

BLUMBE - Best Linear Uniformly Minimum Biased Estimation.

homBLUP - Best homogeneously Linear Unbiased Prediction (related to Ordinary Kriging).

homBLIP - Best homogeneously Linear Prediction (related to Optimal Biased Kriging).

inhomBLIP - Best inhomogeneously Linear Prediction (related to Simple Kriging).

LESS - LEast Squares Solution.

MALE - MAximum Likelihood Estimation.

MSPE - Mean Square Prediction Error.

RV - Random Variable

RF - Random Function.
Common Notations:

In general and unless otherwise mentioned, small Greek characters indicate non-random quantities in contrast to small Latin ones being random.

\* - random estimate of a non-random parameter.

\* - random prediction of a random parameter.

\*' - element associated with the new location to be predicted.

Different predictors:

\( \tilde{x}_r \) - simple kriging, \( \bar{x} \) - ordinary kriging, \( \tilde{x} \) - optimal biased kriging

Greek characters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Denotes</th>
<th>Cressie's (1993)</th>
<th>Name/Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta )</td>
<td>Expectation of a RV also ( E(\mu) )</td>
<td>=</td>
<td>(Beta)</td>
</tr>
<tr>
<td>( \gamma(h) )</td>
<td>Semi-variogram function computed from the RV ( y(s) ) for lag vector ( h )</td>
<td>=</td>
<td>(Gamma)</td>
</tr>
<tr>
<td>( \Gamma )</td>
<td>Semi-variogram matrix of the observation vector ( y ).</td>
<td>=</td>
<td>(Gamma)</td>
</tr>
<tr>
<td>( \eta(h) )</td>
<td>Homeogram function.</td>
<td></td>
<td>(Eta)</td>
</tr>
<tr>
<td>( H )</td>
<td>Homeogram matrix, non-centered variance-covariance matrix of the ( y ).</td>
<td></td>
<td>(Eta)</td>
</tr>
<tr>
<td>( \kappa(h) )</td>
<td>Covariance function.</td>
<td>( c )</td>
<td>(Kappa)</td>
</tr>
<tr>
<td>( K )</td>
<td>Variance - covariance matrix of the observation vector ( y ).</td>
<td>( \Sigma )</td>
<td>(Kappa)</td>
</tr>
<tr>
<td>( \nu )</td>
<td>Coefficient/Vector of Lagrange multiplier/s.</td>
<td>( m )</td>
<td>(Nu)</td>
</tr>
<tr>
<td>( \xi )</td>
<td>Vector of fixed parameters.</td>
<td>( \beta ) (varies)</td>
<td>(Xi)</td>
</tr>
<tr>
<td>Symbol</td>
<td>Denotes</td>
<td>Cressie's Name/Remark</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>----------------------</td>
<td></td>
</tr>
<tr>
<td>$\rho(h)$</td>
<td>Correlation function.</td>
<td>= (Rho)</td>
<td></td>
</tr>
<tr>
<td>$\sigma^2$</td>
<td>Variance component, positive and often unknown.</td>
<td>= (Sigma)</td>
<td></td>
</tr>
<tr>
<td>$\sigma_e^2$</td>
<td>One of the parameters in the variogram model, observation noise variance.</td>
<td>$c_0$ = Nugget (Sigma)</td>
<td></td>
</tr>
<tr>
<td>$\sigma_y^2$</td>
<td>One of the parameters in the variogram model, total observation variance.</td>
<td>$c_0 + c_s$ = Sill (Sigma) $\sigma_y^2 = \sigma_x^2 + \sigma_e^2$</td>
<td></td>
</tr>
<tr>
<td>$\tau$</td>
<td>Summation vector, $[1 1 1 1 . . .]^T$.</td>
<td>1 (Tau)</td>
<td></td>
</tr>
<tr>
<td>$\chi$</td>
<td>Kriging weight coefficients vector.</td>
<td>$\lambda$ (Lambda)</td>
<td></td>
</tr>
</tbody>
</table>

Latin characters

Symbol | Denotes | Cressie's Name/Remark |
--------|---------|----------------------|
| $a_1...a_n$ | Coefficients of a linear function. | |
| $A$ | Matrix of coefficient (also Jacobian matrix of first partial derivatives). | $X$ |
| $c_{si}$ | One of the parameters in the homeogram / covariance function model. | |
| $C_x$ | Dispersion matrix of RV x with the elements $C_x(s_i,s_j)$. | |
| $D(\bullet)$ | Dispersion matrix of vector $\bullet$, also called the variance-covariance matrix. | |
| $e$ | n*1 vector of observational errors. | |
| $E(\bullet)$ | Expectation of vector $\bullet$, mean vector | |
| $h$ | Separation distance or lag. | |
| $n$ | Number of data points. | |

xiv
<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N(h)$</td>
<td>Number of pairs of data values available at lag distance $h$</td>
</tr>
<tr>
<td>$P$</td>
<td>Weight matrix of the observations $y$, which is normally known and positive-definite</td>
</tr>
<tr>
<td>$Q$</td>
<td>Cofactor matrix of $y$; if $P$ is invertible then $Q=P^{-1}$ and $P=Q^{-1}$.</td>
</tr>
<tr>
<td>$r$</td>
<td>One of the parameters in the variogram model (if and where it becomes constant)</td>
</tr>
<tr>
<td>$s$</td>
<td>Spatial location of sample observations.</td>
</tr>
<tr>
<td>$s'$</td>
<td>Spatial location of new point.</td>
</tr>
<tr>
<td>$x(s)$</td>
<td>Vector of continuous RVs of the true process at locations $s \in {s_1, \ldots, s_n}$.</td>
</tr>
<tr>
<td>$X$</td>
<td>East-West spatial coordinate, topocentric frame, pointing east.</td>
</tr>
<tr>
<td>$y(s)$</td>
<td>Vector of continuous RVs of observations at locations $s \in {s_1, \ldots, s_n}$.</td>
</tr>
<tr>
<td>$Y$</td>
<td>North-South spatial coordinate, topocentric frame, pointing north.</td>
</tr>
<tr>
<td>$y$</td>
<td>$n \times 1$ vector of RVs of observations.</td>
</tr>
<tr>
<td>$Z$</td>
<td>Vertical spatial coordinate, topocentric frame, pointing upward.</td>
</tr>
<tr>
<td>$a_s$</td>
<td>Range</td>
</tr>
<tr>
<td>$s_0$</td>
<td></td>
</tr>
</tbody>
</table>

$xv$
CHAPTER 1

INTRODUCTION

1.1 Spatial statistics and GIS

In recent years, there has been a growing interest in statistical methods for spatial prediction; those methods are being applied today in many fields of sciences, such as geodesy, geology and environmental science. Spatial statistical prediction, also called kriging, aims at generating a statistical prediction of an attribute value at unsampled sites from measurements made at point locations within the same area or region. The general rationale behind spatial interpolation is that, on average, values at points close together in space are more likely to be similar than those at points further apart. A frequently asked question is what is the best interpolation method? Naturally, in many cases, the answer to this question depends on the given data set and on the requirements: for example; small MSPE (Mean Square Prediction Error), speed of computation, surface smoothness and continuity etc. Nevertheless, many studies have proved that kriging is a superior prediction method (for example Geoffrey (1994), who compares kriging with splines) since it performs better in terms of MSPE than many other linear methods. Spatial data prediction is frequently carried out using a Geographic Information System (GIS), a collection of hardware and software tools designed to efficiently capture, store, update,
manipulate, analyze, and display all forms of geographically referenced data; spatial, non-spatial, qualitative, and quantitative (ESRI 1995). Cressie and Ver Hoef (1993)

![Diagram of GIS and Geostatistics](image)

**Figure 1.1: Interaction of GIS and geostatistics.**

described a three-layer relationship; the bottom layer is the physical world made up of particles, organisms and their environment. Data from the bottom layer are directed into the middle layer, the GIS, which stores the data and allows their analyses using the top layer, namely the statistical and mathematical models (see Figure 1.1). Consequently, GIS is used as an effective conduit between the top and bottom layers. However, in their paper they noted that the middle layer has yet to be realized. Indeed, much of the research in GIS has been in computer graphics, databases, and computational algorithms. Only
recently have GIS’s begun to include spatial statistical tools and those are still limited (Cressie et al. 1996). The same thoughts appeared also in Dubois (2001) who suggested that recent developments to integrate geostatistics into Geographic Information Systems (GIS) do not seem to have been as fruitful as one could have expected. In his paper he identified events where interactions between GIS and geostatistics exist and consequently the use of GIS can assist in performing geostatistical analysis. Some examples that he mentioned are:

- Validating the logical consistency of a given data set using topological information; e.g., soil samples cannot usually be located within a lake.
- Using the same polygons of exclusion operation in the prediction process to prevent the use of certain measurements.
- Carrying out a multi-criteria and multivariate analysis; e.g. using categorical variables such as land-use polygons to derive local trend of a Digital Elevation Model (DEM) as part of a geostatistical analysis.
- Exploiting GIS visualization tools for exploratory data analysis, analysis of clustering patterns and validation of spatial interpolation results using graphical methods as contour lines, shaded relief maps, three-dimensional views etc.

Griffith and Layne (1999, pp. 475-476) reviewed the latest development in integrating GIS and spatial statistics, and again they repeated the theme that GIS programs are only beginning to assimilate and implement spatial statistical tools. They mentioned the proliferation of specialized packages for geostatistics and review some of their capabilities. It is important to mention that ArcGIS Geostatistical Analyst as described in
Johnston et al. (2001) provides some advanced statistical tools, and it may be the answer for some of the problems reported in the literature (Cressie et al. (1996), Dubois (2001), and Griffith and Layne (1999)). However at the time of the research, we were unable to examine this latest development. In this work, we will look at each layer in Figure 1.1 separately and on the interrelationship between them. In the GIS thread we will look at a faster and computationally more efficient way to search for the interpolation neighborhood. In the statistical thread we will investigate the use of a more efficient predictor to interpolate our data, namely, a predictor with a lower mean square prediction error. Finally, in the “world” layer we will examine those methods using a newly collected aeromagnetic data set from Antarctica, and laser scanner data collected in Ocean City, Maryland. It was not a surprise to see that the statistical approach defines our computational efficiency, and our GIS search strategy influences the accuracy of our prediction. The practical case studies provide a platform to show the link between the two layers and how the theory should be applied taking into account the specific features of our two data sets, being either magnetic data or laser scanner data.

1.2. Research problems

The main objective of this research is to develop advanced interpolation techniques and algorithms that are efficient in terms of their Mean Square Prediction Error (MSPE), i.e. more accurate, and show enhanced computational efficiency. We can categorize this study into five problems:

a) Geostatistical interpolation is a computationally very consuming process. As we will see in Section 2.3, the computation involves the inversion of a $n \times n$ matrix when $n$ is the
number of sampled points. The common way to handle this problem is to include in the kriging process only "nearby" points. Nevertheless, as will be shown in Section 4.1, the choice of the kriging points, sometimes called "support", has a significant effect on the accuracy of our prediction and clearly on the computational needs. Hence, we will investigate the use of Delaunay triangulation structure as a data structure to choose our interpolation neighborhood. The use of Delaunay triangulation structure for interpolation has previously been applied in the case of piecewise polynomial function fitting (Renka and Brown 1999). There are some reports that mention the use of Delaunay triangulation and its dual Voronoi diagram in statistical analysis, for example Nielsen (1994) who emphasized the usefulness of this data structure to avoid under-representation of certain directions. However, he also pointed out that the application of Delaunay triangulation and the dual is still new, and the research did not continue to explore and compare this data structure with respect to other search strategies. Hessami et. al. (2001) compared Delaunay triangulation and fixed radius neighborhood search for kriging, and Watson (1992, pp. 69-73) mentioned it as a method for subset selection. However, a comprehensive analysis of the effect of this special configuration and an algorithm that allows a varying Delaunay neighborhood for prediction is missing.

b) In many applications the map user is interested not in an absolutely accurate map; he/she would prefer a relatively more accurate map. For example: the TAMARA project is an international effort to map the geological structure of the west Antarctic rift system from a newly acquired aero-magnetic data set. Here the scientists are interested in the relative changes over space between magnetic values, which indicate magnetic
anomalies. The absolute magnetic values per point are of minor importance. A comprehensive mathematical theory for the optimal biased predictor with different spatial coherency functions was developed in Schaffrin (1993, 1997, 2000a and 2001). Independently, Gotway and Cressie (1989, 1993) investigated the topic and developed a general form, which uses the covariance matrix, for a large class of predictors, including the optimal biased predictor. Application of the theory on a real world data set was never performed, to our best knowledge, and this will be one of the objectives of this research. Moreover, we will use for the first time the homeogram as our spatial coherency function in the kriging equations and empirically evaluate the advantages and disadvantages of it.

c) With the implementation of Optimal Biased Kriging on massive data sets, we are faced with a computation problem. We cannot use a subset of the data due to the biasedness of this method and, therefore, we should find an efficient method to invert the \( n \times n \) coherency matrix and perform matrix multiplication with it. We propose a new method, which will use spatial sorting to arrange the spatial coherency matrix as a sparse band-limited matrix. Applying this procedure will speed up kriging computation considerably.

d) In the application part, we will apply the theories on newly acquired airborne magnetic data in the West of Antarctica, with more than 270,000 measurement points which cover an area of 150*250km\(^2\). Magnetism as a potential field is a result of the earth magnetic force and the complex geological structure of the ground. Thus it has some unique spatial and statistical properties (see Hansen (1993), and Spector and Grant (1970)), which can be applied in the kriging process to derive a better interpolation results and to get more
information about our research area. During the process of applying the various kriging methods we will examine the computational problems and ways of dealing with them.

Another test data set is from laser altimetry: Scanning air-borne laser systems represent an emerging technology and it is currently an issue for many studies and investigations (e.g. Kraus and Pfeifer 1998, Filin 2001, Schenk 2001a). However, this data set is relatively simple to understand and easy to analyze while yielding a digital elevation model of the topography. We will use the previously developed methods to analyze the spatial dependence function of the data and demonstrate the accuracy of the Optimal Biased Kriging interpolation using this data set.

e) Our study will have an integrated approach in which we weigh different computer algorithms in terms of their computational complexity and statistical properties, and check statistical methods with respect to the procedures to implement them. This will fill a gap that exists in the literature on how to implement geostatistical procedures within a GIS environment.

1.3 Description of our case studies

1.3.1 TAMARA - Aeromagnetic data

Our first case study area is at the Transantarctic Mountains (TAM) in southern Victoria Land, Antarctica, where a joint U.S.-German science project (TAMARA) seeks to resolve some outstanding questions about the evolution of the TAM. Aeromagnetic data were collected in the Antarctic field campaign of 1997-1998 using a Cesium magnetometer flown in a bird slung 30m below an A-Star Helicopter. The helicopter was flying at an average flight height of 300-500 m above the ground to create the so-called
Semi Draped aeromagnetic survey mode. Magnetic reference base stations were established in McMurdo and in the Skelton Neve field camp to record daily variations of the Earth's magnetic field. Data were collected along W/E flight lines separated by 2500 meters and sampled along flight lines at 50 meters interval. Tie lines were flown in N/S direction at 25km separation. Finally, the project's new aeromagnetic data included 270000 points in 180 flight line files that spanned over an area of 150*250 km². These raw total magnetic field data were processed using standard methods - heading and diurnal corrections, leveling and IGRF model subtraction (see British Geological Survey 2001) We would like to interpolate the aeromagnetic data into a grid, in order to integrate it with satellite imagery, existing geological, topographic and other geophysical data. We will focus on a small area of this project at: Xmin=947000, Ymin=948000; Xmax= 992500, Ymax=977500; units are [m].

Treatment of magnetic data. There are many issues pertaining the analysis of magnetic data; in this proposal we will skip the details and just summarize some important facts:

- The magnetic force applied to one magnetic monopole by another magnetic monopole is given by the famous Coulomb's equation.
- Normally, we would have to apply a "reduce-to-pole" filter to get a monopole type field; in our case it is not relevant since our area is near the south pole.
- The magnetic field strength, $H$, is defined as the force per unit pole strength exerted by a magnetic monopole, $pI$.
- Units associated with magnetic field strength are nanoteslas (nT). The average strength of the Earth's magnetic field is about 50,000 nT. A nanotesla is also
commonly referred to as a *gamma*. The Earth's normal magnetic field effect was removed from our data.

- When a magnetic material, say iron, is placed within a magnetic field, $H$, the magnetic material will produce its own magnetization. This phenomenon is called *induced magnetization*.

Figure 1.2: Study area in West Antarctica, on the right of the map the Ross Ice Shelf, and Mt. Discovery, the rectangle at the upper left corner is the study area, both flight lines (light black), boundary line of the TAMARA project and geological lineaments from Wilson (1999) are shown in the figure.
• The strength of the magnetic field induced by the magnetic material due to the inducing field is called the intensity of magnetization, $I$. The intensity of magnetization, $I$, is related to the strength of the inducing magnetic field, $H_f$, through a constant of proportionality, $k_s$, known as the magnetic susceptibility:

$$I = k_s \cdot H_f$$

(1.1)

• The magnetic susceptibility is a unitless constant that is determined by the physical properties of the magnetic material.

By measuring the magnetic field over an area we can infer the type of geologic material that constitutes our area (using 1.1) and learn about the geological structure of it.

Some reports show that kriging is a virtuous method to interpolate aeromagnetic data, for example Hansen (1993). Moreover, there are a few studies in which the covariance was computed analytically by calculating the average effect of an ensemble of many different bodies (see for example Spector and Grant, 1970). However this computation requires prior knowledge of a few geological parameters, e.g. the average depth to the source or the average magnetization per unit volume, and we do not have those parameters.

Therefore we will compute our spatial coherency function empirically.
1.3.2 Ocean City- Laser scanning data

NASA's Goddard Space Flight Center (GSFC) is currently developing a new laser scanning system. Mounting the laser on a P-3 aircraft and flying it over the Wallops/Ocean City area will validate the system performance. The photogrammetry group at Ohio State University with Bea Csathó and Toni Schenk as principle investigators was assigned to create a dense and accurate Digital Elevation Model (DEM) of the area, which will be used for the validation process (see Schenk 2001b). The source datasets to create this precise DEM were collected by an Airborne Topographic Mapper laser scanner system. We chose two of those freely available datasets, which were collected on Assateague Island, an unsettled area with mostly flat topography (see Figure 1.2, and Krabill et al. 2000 for full description of the data). The simplicity of the surface and of the underlying process will be ideal for testing our statistical models. The developed method may serve later for the DEM creation, although at this stage we applied a thin-out process by taking only one point out of 80 and cutting out a small portion of the area (630m*750m). The technical specifications of our original data are summarized in Table 1.1 with some computed parameters using Baltsavias (1999):
<table>
<thead>
<tr>
<th>Data collected in</th>
<th>October 1999</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flight height</td>
<td>610m above ground level</td>
</tr>
<tr>
<td>Laser pulse rate</td>
<td>5kHz</td>
</tr>
<tr>
<td>Scanner rotation frequency</td>
<td>20Hz</td>
</tr>
<tr>
<td>Off-nadir angle</td>
<td>15 degrees (30-degree swath width)</td>
</tr>
<tr>
<td>Typical flight speed</td>
<td>120 knots, or 61.7m/sec</td>
</tr>
<tr>
<td>Footprint diameter</td>
<td>~0.6m</td>
</tr>
</tbody>
</table>

**Computed parameters**

| Laser swath width         | \(2 \times 610 \times \sin 15^\circ = 315m\) |
| Number of points per cycles | \(5000Hz / 20Hz = 250\) |
| Distance between consecutive scans | \(61.7 / 20 = 3.085m\) |
| Average distance between footprints centers | \(2 \times \pi \times 157 / 250 = 3.945m\) |

**Table 1.1: Specification of the laser scanner data.**

Note that the scan geometry causes a greater point density at the edges of the swath than at the center. The original files had 91521 points; after cutting out overlapping areas and the thin-out process, we got 1099 points as input data and a set of independently measured 331 points as validation points.
1.4 Organization of the report

In general, I have endeavored to give a comprehensible presentation of the theory and practice of the geostatistical and computational geometry techniques which are used in this work. However, it was sometimes impossible within the scope of this treatise, to detail every method used, and therefore I gave a wide list of references for any topic not discussed in details. This dissertation is organized in six parts in accordance with the motivations and issues brought out above. The first part (Chapter 1) introduces the research background, case studies and objectives. Chapter 2 provides a short literature review on the wide topic of interpolation methods, after a brief introduction to geostatistics it describes spatial coherency function analysis and uses the developed methodology and algorithms to derive the functions for the data of the case study areas. Last the geostatistical paradigm is formulated, i.e. the mathematical model is defined and equations for simple and ordinary kriging interpolation are derived.
Chapter 3 looks into computational geometry algorithms and data structures that are being used for spatial statistics analysis. We outline the algorithms for neighborhood search; we continue with two fundamental spatial data structures, the Delaunay triangulation and Voronoi diagram. After an analysis of the properties of those data structures we identify their usefulness for spatial statistics. The computational complexity of neighborhood search algorithms is an important factor in designing and programming of GIS modules. Consequently, in this chapter we study efficient algorithms that can solve the neighborhood search task, and finally we describe a Delaunay data structure algorithm as a search method for spatial statistics interpolation.

Chapter 4 addresses the problem of neighborhood selection from a statistical viewpoint. We summarize some statistical examples in which the notion of neighborhood selection exists. We then examine the effect of data configuration on the interpolation results. It uses Fourier analysis to quantify sufficient neighborhoods for kriging interpolation. Finally, we report on the results of applying our Delaunay structured program to the TAMARA case study and the Ocean City case study.

Chapter 5 investigates a method to improve the interpolation MSPE. This method, named here Optimal Biased Kriging, is based on the homBLUP prediction technique. First, we derive the basic formulas and explain the concept, then we discuss an important implementation issue namely inverting the spatial coherency function, and last we present the results of applying this algorithm to interpolate topographic digital elevation data acquired by an airborne laser scanner system. Chapter 6 concludes the research and indicates some problems and topics that should be further investigated.
CHAPTER 2

GEOSTATISTICAL INTERPOLATION METHODS

This chapter starts with a short review of spatial interpolation methods and with information needed to classify the different techniques. In this review, we also present the geostatistical interpolation process (kriging). Explanatory data analysis is an essential part of kriging, therefore a thorough study of this topic is given in this chapter. In this study we derive the spatial coherency functions of our test data using our own program. Finally we give the details of kriging: the mathematical model and its assumptions, the prediction solutions of simple and ordinary kriging and their computational forms.

2.1 Review of interpolation methods

2.1.1 Classification of methods

Many interpolation methods have been developed in the recent years. Burrough and McDonnell (1998, p. 103) grouped interpolation methods into global and local methods. Global interpolation methods exploit all the sample points in predicting values at new positions. They are often used to model the regional trends (long range variation) and not so suitable for direct interpolation since local variations are often suppressed. Global methods include classification methods, regression or trend analysis (see Section 2.1.3), Fourier transform and, to some extent, wavelets. Once the global effects have been taken care of, the residuals from the global variation can be interpolated locally. Local
interpolation algorithms utilize a small zone around the point being interpolated and include methods such as: Voronoi polygons (see Section 3.3), linear interpolation (see Delaunay triangulation in Section 3.2), conditional stimulation (used also as a global method), zonal kriging, and minimum curvature interpolation. A summary table of those interpolation techniques is given in Burrough and McDonnell (1998, pp. 158-159) and the following properties are used for their classification: deterministic or stochastic methods, global and local methods, abrupt or gradual transitions, exact interpolator which honors the original data points, computing load and the assumptions of the model. Cressie (1989) divided spatial prediction methods into stochastic methods which include: simple kriging, ordinary kriging, universal kriging, kriging with intrinsic random functions, Markov random field prediction, transgaussian kriging, disjunctive kriging and Bayesian nonparametric smoothing while nonstochastic methods include: moving average, inverse distance, Delaunay triangulation, natural neighbor interpolation, splines and multiquadric-biharmonic interpolation. Sárkozy (1998) categorized the methods as: Method of geometrical nearness; Statistical methods based on weighted average; Methods using basis functions; Methods of artificial neural networks. Mathematically, the interpolation problem is defined as follows:

Suppose that our measurements are denoted by:

\[ \{ y(s_i): s_i \in D; \ i=1, \ldots, n \} \tag{2.1.1} \]

where \( s_i \) denotes a spatial location in \( \mathbb{R}^2 \). \( D \) is the extent of the region of interest and \( s \) varies continuously across it. Our observations are assembled in the \( n \times 1 \) vector:

\[ y=[y(s_1), \ldots, y(s_n)]^T \tag{2.1.2} \]
measured with observational noise at known sites \( \{s_1, s_2, \ldots, s_n\} \). We would like to compute the actual value \( x(s') \) by predicting \( \bar{x}(s') \) or estimating \( \hat{x}(s') \) at an arbitrary location \( s' \in D \). In the next sections we will give an overview of a few methods to solve this problem.

2.1.2 Inverse distance weighting

The method is characterized as deterministic, local, gradual, not exact, best for quick interpolation, with very small computing load and with no initial assumption. The new value at the unknown location is determined using a linearly weighted combination of observations taken from a set of sample points. The weight is chosen as a function of the inverse distance.

\[
x(s') = \frac{\sum_{i=1}^{n} \frac{1}{d_{s_i}^p} \cdot y(s_i)}{\sum_{i=1}^{n} \frac{1}{d_{s_i}^p}} = \left[ \frac{1}{d_{s_1}^p} \ldots \frac{1}{d_{s_n}^p} \right] \left[ \begin{array}{c} y(s_1) \\ \vdots \\ y(s_n) \end{array} \right] \left[ \begin{array}{c} 1 \\ \vdots \\ 1 \end{array} \right]
\]

where \( p \geq 1 \) is the power of the inverse distance, mostly \( p \in \{1, 2, 3\} \). The interpolation can be shifted from local to global by changing the power value. A larger power value will result in less influence from surrounding points, i.e., nearby data will have the most influence. To compute the Mean Square Error we can rewrite (2.1.3.1) as:

\[
D \{ \hat{x}(s') \} = MSE \{ \hat{x}(s') \} = (\tau^T D^{-1} \tau)^{-2} \cdot (\tau^T D^{-1} \cdot D \{ y \} \cdot D^{-1} \tau)
\]
Limiting the input points for calculating each interpolated point can also control the characteristics of the interpolated surface. The input can be limited by the search option (see Section 3.1), for example by identifying the number of sample points used, or a radius within which all points are used in the calculation of the interpolated points. The best results from Inverse Distance Weighting (IDW) are obtained when sampling is sufficiently dense with regard to the local variation that we attempt to simulate. If the sampling of input points is sparse or very uneven, the results may not sufficiently represent the desired surface.

2.1.3 Trend surface analysis

This technique is deterministic, global, gradual, not exact, best for removal of trend, with small computing load, allows a phenomenological explanation of the data and assumes that the surface is smooth. The trend surface interpolator uses a least-squares adjustment to fit a surface to the input points. Normally, with the trend interpolator the user should select the order of the polynomial used to fit the surface. As the order of the polynomial is increased, the surface being fitted becomes progressively more complex and less smooth. A higher order polynomial will not always generate the most accurate surface. The lower the Mean Square Error (MSE), the more closely will the interpolated surface represent the input points. The most common order of polynomials is 1 through 3; for example, in this research we used a 3rd order polynomial:

\[ Z_i = a_0 + a_1 x_i + a_2 y_i + a_3 x_i y_i + a_4 x_i^2 + a_5 y_i^2 + a_6 x_i^2 y_i + a_7 x_i y_i^2 + a_8 x_i^3 + a_9 y_i^3 + e_i \]
\[
Z_i = \begin{bmatrix} 1 & X_i & Y_i & X_i^2 & Y_i^2 & X_i^2Y & X_iY_i & X_i^3 & Y_i^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_9 \end{bmatrix} + e_i = \alpha(s')^T \cdot a + e_i
\]

for \(i=1,\ldots,n\) \((n>10)\)

\[
D\{z(s')\} = D\{\hat{x}(s')\} = MSE\{\hat{x}(s')\} = \alpha(s')^T \cdot D(\hat{a}) \cdot \alpha(s')
\]

\(Z_i\) is the observation, \(X_i\) and \(Y_i\) are the planar location coordinates, and \(a_1,\ldots, a_9\) are the coefficients which will be computed in the adjustment process by minimizing the sum of squared errors \(e_i\) (see the least-squares 1D example in Section 4.2). Trend surface interpolation tends to create smooth surfaces. The surface generated will seldom pass through the original data points since it performs a best fit for the entire surface.

2.1.4 Minimum curvature interpolation

As the trend surface method, minimum curvature interpolation uses basis functions fitted to the sample points. It is deterministic, local, gradual, not exact, best for DEM's with moderately detailed data, with small computing load, and assumes that the surface is smooth. A minimum curvature interpolation results in a smooth surface that passes exactly through the relevant neighborhood of the input points. The basic minimum-curvature technique is also referred to as thin plate spline interpolation since the basis function also solves the differential equation of a thin, linearly elastic plate passing through each of the data values with a minimum amount of bending. The Minimum Curvature algorithm produces a grid and repeatedly computes the grid points values by fitting functions that will solve the differential equation on a relevant grid points neighborhood. The fitting process includes boundary condition equations that ensure a
smooth (continuous and differentiable) surface together with continuous first-derivative surfaces. We take for example one type of minimum curvature interpolation that uses the following formula for the surface interpolation:

\[ \hat{x}(s') = T(s') + \sum_{j=1}^{n} a_j \cdot R(d_j) \]  \hspace{1cm} (2.1.5)

\[ \sum_{j=1}^{n} a_j = 0 \]

where:

\( \hat{x}(s') \) is the interpolated value,

\( j = 1, 2, ..., n \) is the number of points,

\( a_j \) are coefficients found by the solution of a linear system,

\( d_j \) is the distance from the point \( s' \) to the \( j^{th} \) point \( s_j \).

\( T(s') \) and \( R(d_j) \) are defined according to the type of minimum curvature method; for example, in the so-called "Tension" method we have:

\[ T(s') = a_0 \]

\[ R(d_j) = \frac{1}{2\pi \varphi^2} \left[ \ln \left( \frac{d_j \varphi}{2} \right) + c_1 + K_0(d_j, \varphi) \right] \]  \hspace{1cm} (2.1.6.1)

\( \varphi \) is the weight tension parameters entered by the user.

\( K_0 \) is the modified Bessel function of first kind with zero order.

\( c_1 \) is the Euler's constant equal to 0.577215.

The coefficient \( a_0 \) is found using least squares adjustment along with \( a_1, ..., a_n \); thus the system has \( n+1 \) unknowns and \( n+1 \) equations, and we can write:
\[ \hat{x}(s') = [1 \quad R(d_1) \quad \ldots \quad R(d_n)] \begin{bmatrix} \hat{a}_0(s') \\ \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix} = \gamma^T \cdot \hat{a} \quad \gamma = \gamma(s') \]

and

\[ D\{\hat{x}(s')\} = MSE\{\hat{x}(s')\} = \gamma^T \cdot D\{\hat{a}\} \cdot \gamma \quad (2.1.6.3) \]

There are many variants and algorithms for minimum curvature interpolation. They differ on the way the local neighborhood is selected and the way the boundary conditions are imposed; for example, the regularized option of minimum curvature interpolation works on the same principles, but with a more complicated expressions for \( T(s') \) and \( R(d_j) \) (for more details we refer to Briggs 1974, Smith and Wessel 1990, or Franke 1982).

### 2.1.5 Area stealing interpolation

This method uses geometrical nearness as an interpolation criterion; it is often called 'natural neighbor interpolation' since the weights are determined using the Voronoi tessellation. In Figure 2.1 we constructed the Voronoi cells for the sample points \( P_1 \ldots P_5 \) with measured values \( y_1 \ldots y_5 \). We want to find the surface value \( \hat{x}(s') \) at the given location \( s' \) by interpolation. For this purpose, we have to redefine the tessellation with the addition of the interpolation point \( s' \). Now we have two tessellations the original one (without the new point) and the new tessellation constructed after inserting the new point. The tile( cell) of the new point covers some parts of the tiles originally belonging to particular sample points. These particular points will be involved in the interpolation of the new point.
The weights of the natural neighbors are nothing else but the areas, which the new cell cuts out from the original cell, owned by a particular neighbor. These areas are the 'stolen areas', denoted in the formula (2.1.7) with $V_i$

$$\hat{x}(s') = \sum_{j=1}^{n} a_j(s') \cdot y(s_j) = \alpha^T y \quad , \quad \alpha = \alpha(s')$$

where

$$a_j(s') = \frac{|V_i(s')|}{|V(s')|}$$

and

$$D(\hat{x}(s')) = MSE(\hat{x}(s')) = \alpha^T \cdot D\{y\} \cdot \alpha$$

$V(s')$ is the area of the Voronoi polygon around the new point, $V_i(s')$ is that part of $V(s')$ which belongs to the tile of point $s_i$. This method automatically selects from the sample points those which should take part in the interpolation. These points are called natural neighbors. Notice that, when the new point reaches the observation location $s_i$, then $a_i = 1$, while all the other observations are assigned zero weights. Sibson (1981) defined (2.1.7)
as natural neighbor $C^0$ interpolant since the derivative of it is discontinuous at the data sites. He proposed the $C^1$ natural neighbors interpolation which is a mixture of (2.1.7) and a polynomial function that is tailored to have the properties of continuity and differentiability even at the data points. Gold (1989) proposed to use a simple blending function to make the surface "smoothly" continuous at data points (i.e. differentiable).

2.1.6 The geostatistical paradigm (Kriging)

Kriging is a stochastic interpolation method in contrast to the deterministic methods reviewed in Sections 2.1.2-5. Kriging is derived from regionalized variable theory which assumes that the spatial variation of any geophysical, soil, or hydrological property, known as a regionalized variable, is statistically homogenous throughout the surface, that is, the same pattern of variation can be observed at all locations on the surface. This spatial variation of the property can be expressed in terms of the spatial coherency function (Semi-variogram, Covariance function, Homeogram). The geostatistical interpolation process consists of four steps (see also Dubois 2001):

- Explanatory data analysis, by which we actually need to learn about the characteristics of our data. Specifically, we probe for outliers or inconsistencies, and we assess if there is a trend. (If there is, then we need another step to remove the trend and/or the outliers.)

- Calculation of the spatial coherency function and fitting a mathematical model to it.
• The actual **Kriging process** can be considered as a weighted average technique, which uses the spatial coherency function to obtain the relationship between the data points.

• Validation of the interpolation results.

The computation and interpretation of a semi-variogram is the “heart” of the kriging method. Semi-variograms measure the lack of spatial correlation among observational data points in a planar study area as a function of the distance and direction between observational data points. They control the way in which kriging weights are assigned to data points during interpolation, and consequently control the quality of the results. We will give a detailed explanation about this process in Section 2.2.

Kriging overcomes many shortcomings of the previously reviewed interpolation methods. For instance, the kriging weights are determined by the spatial coherency function and the configuration of the data set. Since the prediction variances can be determined and mapped, by assuming a particular probability distribution, we can calculate the confidence that we can place in the prediction. This makes kriging uniquely different from the above mentioned interpolation methods which essentially neglect measures of spatial coherency. The variance of the prediction can also be used to determine where more information is needed if future sampling is planned. Kriging weights depend not only on the distances between observed and predicted locations, but on the mutual distances among observational points as well. As a result, kriging has two interesting and unique properties: a declustering and a screen effect (Hu 1995). With the declustering property, a few observations close together will have collectively the weight of a single
observational point located near the centroid of the cluster. With screen effect, the influence of an observational point will be reduced by addition of one or more observational points at the intermediate locations between the originally observed and the predicted location (see Chapter 4 for a related discussion). Like any other method, kriging has a few deficiencies. Due to the stationarity assumption, kriging is not a suitable method for data sets which have anomalous pits or spikes, or abrupt changes such as breaklines. Kriging computation is also demanding, and efficient algorithms should be developed to perform those computations on a big data sets (see Chapter 3 and Section 5.2 for further discussion).

2.2 Estimation of spatial coherency functions

2.2.1 Introduction and basic stochastic assumptions

Spatial continuity/coherency exists in most data sets from earth sciences. Two data points close to each other are more likely to have similar values then two data points that are far apart. This association is known for many years and was already described by Fisher (1935). However, statistical models to use this phenomenon did not begin to appear until much later (i.e. early 50's; see also Cressie 1990). Proper spatial dependence analysis is a vital step in the geostatistical analysis; this topic has many applications and can fill up an entire book (e.g. Pannatier 1996). However, studies have shown that the influence of an error there amounts only to a second order effect in the precision to which the optimal solution is determined. Moritz (1976, p. 70) was the first to analyze and demonstrate this fact, and Chilès and Delfiner (1999, p. 175 ) wrote: "It takes a gross misspecification of the Variogram model to have a dramatic impact on kriging estimates"; (see also Press et
al. 1992, p.548 on the optimal Wiener filter). For that reason, in this work, we will only summarize the essential concepts and current statistical tools that are used to compute the spatial dependency function. We will start by defining some conditions that will allow us to analyze the data. Following Journel and Huijbregt (1978, pp. 30-35) we let

\[ \{ x(s) : s \in D \subseteq \mathbb{R}^d \} \]  

be a real-valued stochastic process defined on a domain D of the d-dimensional space \( \mathbb{R}^d \), and \( h \) be a distance difference vector; for example: when \( d=2 \) then

\[ |h| = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2}. \]

Notice: For this section we define our functions using the true process \( x(s) \). From Section 2.3 onwards and in practice we only have the observations \( y(s) \), and so we transfer the same definitions to the observations vector.

**Strict Stationarity** - A RF is said to be strictly stationary if its statistical properties are invariant under translations, i.e. two RVs \( x(s) \) and \( x(s+h) \) have the same statistics for any translation vector \( h \). In practice, it is often enough to constrain only the first two moments.

**Stationarity of second order** - A RF is second order stationary iff

(i) \[ E\{x(s)\} = \beta \quad \forall s \in D; \]  

the expectations exist and does not depend on the location.

(ii) for each pair of RVs the covariance exists and depends on the separation distance vector \( h \):

\[ C(h) = E\{x(s+h)\cdot x(s)\} - \beta^2 \quad \forall s, s+h \in D. \]
The stationarity of the covariance implies the stationarity of the variance and of the semi-
variogram - \( \gamma(h) \). The following relations are immediately evident:

\[
\text{Var} \{x(s)\} = E \{[x(s) - \beta]^2\} = C(0) \quad \forall s \in D, \quad (2.2.4)
\]

\[
\gamma(h) = \frac{1}{2} E \{ [x(s+h) - x(s)]^2 \} = C(0) - C(h) \quad \forall s, s+h \in D. \quad (2.2.5)
\]

This second order stationarity is also called **Homogeneity** in the geodetic science
literature.

**Isotropy:** The above conditions still allow different behavior for different spatial
directions; hence we introduce an additional direction invariance condition:

\[
2\gamma(h) = 2\gamma^o(||h||) \text{ for } h \in \mathbb{R}^d \quad (2.2.5.1)
\]

which means that the spatial dependence function is a function of only the distance
between sites, not of their relative orientation. This type of variogram that fulfills
(2.2.5.1) is said to be isotropic; otherwise it is said to be anisotropic. This notation is used
for all the different spatial continuity/coherency functions (Covariance, Homeogram,
Correlogram). A simple form of anisotropy is called geometric anisotropy; for which

\[
\gamma(h) = \gamma^o((h \cdot B \cdot h)^{1/2}), \text{ where } B \text{ is a positive-definite matrix.}
\]

**Intrinsic Stationarity:** A RF \( x(s) \) is said to possess intrinsic stationarity when:

(i) \( E\{x(s)\} = \beta \quad \forall s \in D; \quad (2.2.6) \)

the expectation exists and does not depend on the location, equivalently described by the
first difference \( E( x(s+h) - x(s) ) = 0. \)

(ii) for every distance vector \( h \) the difference \( [x(s+h) - x(s)] \) has a finite variance which
does not depend on \( s \):

\[
\text{Var} \{x(s+h) - x(s)\} = E \{[x(s+h) - x(s)]^2\} = 2\gamma(h) \quad \forall s, (s+h) \in D. \quad (2.2.7)
\]
Quasi-Stationarity: We use the quasi-stationarity hypothesis when we limit our stationarity assumptions to a given distance $|s-s'| < \text{Max\_dist}$.

The type of stationarity and its existence depend on the chosen resolution, and thus, by limiting the distance or our neighborhood points. We can practically impose stationarity.

Ergodicity: A central theme in the application of stochastic process theory is the estimation of statistical parameters in terms of the observed data. We want to express those parameters as expected values of some functional of a process $x(s)$.

A spatial process is called ergodic if its characterizing information is completely contained in any of its realizations. Consequently, for an ergodic process, the expectation over all possible realizations is equal to the spatial average over a single unbounded realization (Schaffrin 1999). Ergodic processes are always second order stationary; the opposite is not necessarily true (for example on a sphere). With the notion of ergodicity we can estimate the statistical moments of the RF from its corresponding mean values computed from the observed data; Papoulis (1991, pp. 427-442) gives more details about ergodicity.

2.2.2 Estimation of spatial coherency functions

The covariance function:

The covariance function is a measure of similarity between the RVs at different locations and, under the assumption of ergodicity, can be estimated by

$$C\{x(s), x(s')\} = \text{E}\{(x(s) - \text{E}(x(s)))[x(s') - \text{E}(x(s'))]\} = (2.2.8)$$
\[ N(h) = \frac{1}{N(h)} \sum_{i=1}^{N(h)} \{ x(s_i) + h \cdot x(s_i) \} - \hat{\beta}^2 \]

where \( \hat{\beta} = \frac{1}{n} \sum_{i=1}^{n} x(s_i) \)

\( N(h) \) is the number of point-pairs separated by the distance \( h \). Averaging over intervals of the magnitude of vector \( h \) — i.e. creating distance or lag classes — is done to obtain a sufficiently high number \( N(h) \) to ensure a small estimation variance (the estimation variance is proportional to \( 1/N(h) \)). Averaging over intervals of the argument of \( h \) — i.e. creating angular classes — is done to check for anisotropy.

**The variogram:**

The variogram function is a measure of dissimilarity between the RVs at different locations. We will estimate half of the variogram or the semi-variogram under the assumption of ergodicity:

\[
\gamma(h) = \frac{\sum_{i=1}^{n} (x(s_i) - x(s_i + h))^2}{2N(h)} = \frac{D\{x(s) - x(s+h)\}^2}{2} = \frac{\sum_{i=1}^{n} (x(s_i) - x(s_i + h))^2}{2N(h)}
\]

\[
\gamma(h) = \frac{E\{x(s)^2\} - 2E\{x(s)x(s+h)\} + E\{(x(s+h))^2\}}{2} = \frac{2E\{(x(s) - \beta)^2\} - 2(E\{x(s)x(s+h)\} - \beta^2)}{2} = \frac{2\sigma_x^2 - C(s,s+h)}{2} = C(0) - C(h)
\]

where \( \sigma_x^2 = \text{Var}\{x(s)\} \) is the variance of \( x(s) \) at any arbitrary location \( s \) (or \( s+h \)), and

\( n \) = number of sample points. In (2.2.9) we used the assumption of ergodicity to replace the expectation by the spatial average. When we use the variogram we can assume the more general assumption of Intrinsic Stationarity, in which case only the first line of (2.2.9) may be true, and we may have to employ another estimation method, for example
using Maximum Likelihood. It is important to note that a particularly robust method to estimate the variogram has been proposed in Cressie (1993, pp. 74-77).

The correlation function:

This tool is defined under the second order stationarity assumption as

\[ \rho(s, s+h) = \frac{C(s, s+h)}{\sigma_x(s) \cdot \sigma_x(s+h)} = \frac{C(h)}{\sigma_x^2} = \frac{C(h)}{C(0)} = 1 - \frac{\gamma(h)}{C(0)} \]  

(2.2.10)

The correlation is unitless (see Isaaks and Srivastava 1989, pp. 30-32)

The homeogram:

Also known as non-centered covariance function, the homeogram, like the covariance function itself, represents a measure of similarity:

\[ \eta(s, s+h) = E\{x(s) \cdot x(s+h)\} = C(h) + \beta^2 \approx \frac{1}{N(h)} \sum_{i=1}^{N(h)} x(s_i) \cdot x(s_i + h) \]  

(2.2.11)

Similar to the variogram, in the homeogram we do not need to implement an estimate for the expectation \( \beta \).

The madogram

The madogram is similar to the variogram, except that the absolute difference between \( x(s) \) and \( x(s+h) \) is calculated rather than the square of the difference as in the traditional semi-variogram. The formula thus becomes

\[ M(h) = \frac{\sum_{i=1}^{N(h)} |x(s_i) - x(s_i + h)|}{2N(h)} \]  

(2.2.12)
The madogram is less sensitive to outliers than the variogram, the covariance function, or the homeogram. It can be useful for inferring range and anisotropy for data sets with extreme data values that would otherwise make variogram features hard to discern.

2.2.3 Variogram Vs. Covariance function Vs. Homeogram

In this section, we will review comparisons made between the three main spatial coherency functions presented above. We saw in Section 2.2.2 that the variogram is defined in cases when the covariance function does not exist (i.e. processes which are only intrinsically stationary). Cressie and Grondona (1992) proved that variogram estimation is to be preferred over covariance function estimation, the two main reasons for that are:

- Both the variogram estimator and covariance function estimator given in (2.2.9) and (2.2.8) lead to generally biased Kriging prediction. However, the variogram based bias tends to be of smaller order.

- If our data have trend contamination then this has “disastrous effects” on attempts to estimate the covariance function (effect of an $O(n^2)$ additive term) while on the variogram it has a “small upward shift”.

The covariance function is sometimes used because computations in the Kriging algorithms are more efficiently performed and are more stable; see U.S.A.C.E (1997, pp. 4-12) and Section 5.2. Kriging that uses the homeogram function as structural dependence function has never been performed as far as we know. Jeannée and de Fouquet (2000) used the homeogram to estimate the variogram and performed kriging by using this variogram on soil pollutant data. They (and also Srivastava and Parker 1989)
claimed that the variogram lacks robustness in the case of positively skewed distributions of the data. Moreover, they maintain that even though variogram fluctuations are rather small at short distances in the Gaussian model, these can become huge in the non-Gaussian case while they are sometimes smaller for the non-centered covariance. They used the relation

\[ \gamma(h) = \frac{1}{2} E \{[x(s) - x(s + h)]^2 \} = \frac{1}{2} E[x(s)^2] + \frac{1}{2} E[x(s + h)^2] - E[x(s)x(s + h)] \]  

(2.2.13)

and from (2.2.11) under stationarity

\[ \gamma(h) = \eta(0) - \eta(h) \]  

(2.2.14)

However, they suggest that care has to be taken with the non-centered covariance since, in case of non-stationarity, its results cannot be trusted. Accordingly, they argue that using the homeogram independently from the variogram should be avoided. Another reason to support this argument is the trend contaminating effect which is shown in Cressie (1993, p. 72) to have a critical influence on covariance estimation and, using the same formulas, we can see that it has equivalent impact on the homeogram estimation. To conclude this review, it seems that the variogram should be preferred over covariance function, and in the case of stationarity (with no trend) the homeogram should be favored.

2.2.4 Fitting a model to the spatial coherency functions

We would like to summarize spatial coherency in a few parameters. This process basically means fitting a nonlinear function to our spatial dependence diagram (variogram, covariance function, homeogram function.) Table 2.1 lists a few common isotropic variogram/covariance function models. The following parameters are used for those models (with the names as used in the geostatistical jargon/literature):
Range - denoted by $r$: The distance at which the semi-variogram reaches 95% of a plateau where an increase in the separation distance no longer causes a corresponding increase in the semi-variogram.

Nugget - denoted by $\sigma_e^2$: Nugget variance is the y-intercept of the semi-variogram model. It represents a variation that is not spatially dependent over the range examined such as the measurement error variance of the observation vector $y$.

Sill - denoted by $\sigma_y^2$, the value of the semi-variogram plateau as found at locations separated by a distance beyond which the semi-variogram does not change, consequently indicating that they are spatially independent of one another. Theoretically the sill of the observed data is best interpreted as the sample variance $\sigma_y^2 = \sigma_x^2 + \sigma_e^2$. The formulas for the homeogram and the covariance function use another parameter $c_{ni}$ which is a constant shift of the graph above the zero level, this parameter reflects the mean of our data in the homeogram or a residual mean which was not captured by the covariance function. The mathematical models for (2.2.16-20) in Table 2.1 are adapted from Cressie (1993, pp. 61-63) with sill $\sigma_y^2 > 0$, nugget $\sigma_y^2 > \sigma_e^2 \geq 0$, range $r > 0$, slope $b \geq 0$. Model (2.2.19) does not possess the second order stationarity property and consequently the covariance function is not defined. The required provision $0 \leq \lambda < 2$ is needed here to satisfy the variogram condition (see Cressie 1993, p. 62). Formulas (2.2.14) and (2.2.15) are taken from U.S. Army Corps of Engineers (1997, pp. 2-8). Formula (2.2.22) is taken from Handcock and Wallis (1994) where $\Gamma$ is the gamma function and $K_{\nu_2}$ is the modified Bessel function of the second kind and order $\nu_2$ (applied by using Matlab $\text{besselk}$ function); $\theta$, $\theta$, and $\alpha$ are arbitrary parameters that are defined in the model fitting part.
<table>
<thead>
<tr>
<th>Model</th>
<th>Expressions for $\gamma(h)$</th>
<th>Expressions for $C(h)$ or $\eta(h)$</th>
<th>No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>$\sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[ 1 - \exp\left( -\left( \frac{h}{r} \right)^2 \right) \right]$</td>
<td>$c_{st} + \sigma_y^2 \cdot \left[ \exp\left( -\left( \frac{h}{r} \right)^2 \right) \right]$</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>$0, h = 0</td>
<td>$c_{st}, h = 0$</td>
<td>.15</td>
</tr>
<tr>
<td>Exponential</td>
<td>$\sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[ 1 - \exp\left( -\left( \frac{h}{r} \right) \right) \right]$</td>
<td>$c_{st} + \sigma_y^2 \cdot \left[ \exp\left( -\left( \frac{h}{r} \right) \right) \right]$</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>$0, h = 0</td>
<td>$c_{st}, h = 0$</td>
<td>.16</td>
</tr>
<tr>
<td>Spherical</td>
<td>$\sigma_y^2 \left( 1 - \frac{h}{r} \right)^6 \left[ 1 + \left( \frac{h}{r} \right)^3 \right]$</td>
<td>$c_{st} + \sigma_y^2 \left[ 1 - 1.5 \left( \frac{h}{r} \right) + 0.5 \left( \frac{h}{r} \right)^3 \right], 0 \leq h \leq r$</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>$0, h = 0</td>
<td>$c_{st}, h &gt; r$</td>
<td>.17</td>
</tr>
<tr>
<td>Rational</td>
<td>$\sigma_y^2 \left( 1 - \frac{h}{r} \right)^6 \left[ 1 + \left( \frac{h}{r} \right)^3 \right]$</td>
<td>$c_{st} + \sigma_y^2 \left[ 1 - 1.5 \left( \frac{h}{r} \right) + 0.5 \left( \frac{h}{r} \right)^3 \right], 0 \leq h \leq r$</td>
<td>2.2</td>
</tr>
<tr>
<td>Quadratic</td>
<td>$\sigma_y^2 \left( 1 - \frac{h}{r} \right)^6 \left[ 1 + \left( \frac{h}{r} \right)^3 \right]$</td>
<td>$c_{st} + \sigma_y^2 \left[ 1 - 1.5 \left( \frac{h}{r} \right) + 0.5 \left( \frac{h}{r} \right)^3 \right], 0 \leq h \leq r$</td>
<td>.18</td>
</tr>
<tr>
<td>Linear</td>
<td>$\sigma_e^2 + bh \sigma_y \</td>
<td>h</td>
<td>\left[ 1 + \left( \frac{h}{r} \right) \right]$</td>
</tr>
<tr>
<td>with sill</td>
<td>$0, h = 0</td>
<td>$c_{st}, h = 0$</td>
<td></td>
</tr>
<tr>
<td>Linear</td>
<td>$\sigma_e^2 + bh \sigma_y \</td>
<td>h</td>
<td>\left[ 1 + \left( \frac{h}{r} \right) \right]$</td>
</tr>
<tr>
<td>Power</td>
<td>$\sigma_e^2 + bh \sigma_y \</td>
<td>h</td>
<td>\left[ 1 + \left( \frac{h}{r} \right) \right]$</td>
</tr>
<tr>
<td>Wave</td>
<td>$\sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[ 1 - \left( \frac{h}{r} \right)^2 \sin\left( \frac{h}{r} \right) \right]$</td>
<td>$c_{st} + \sigma_y^2 \left[ 1 - \left( \frac{h}{r} \right)^2 \sin\left( \frac{h}{r} \right) \right]$</td>
<td>2.2</td>
</tr>
<tr>
<td>Function</td>
<td>$0, h = 0</td>
<td>$c_{st}, h = 0$</td>
<td>.21</td>
</tr>
<tr>
<td>Matérn Class</td>
<td>$-</td>
<td>$c_{st} + \frac{\alpha}{2(\theta_2-1) \Gamma(\theta_2)} \left( \frac{h}{\theta_1} \right)^{\theta_2} K_{\theta_1} \left( \frac{h}{\theta_1} \right)$</td>
<td>2.2</td>
</tr>
<tr>
<td>of Functions</td>
<td></td>
<td></td>
<td>.22</td>
</tr>
</tbody>
</table>

Table 2.1: Some mathematical models for semi-variogram/covariance function with lag distance $h$, sill $\sigma_y^2 > 0$, nugget $\sigma_y^2 - \sigma_e^2 \geq 0$, range $r > 0$, slope $b \geq 0$, $0 \leq \lambda < 2$ (see also Figure 2.2.1.)
Figure 2.2: Examples for semi-variogram (Solid line) and covariance function/homeogram (dashed line) functions where we chose Sill=600; range=500; Nugget=1; c_d=0;

a) Exponential; Table 2.1 formula 2.2.16

b) Gaussian; Table 2.1 formula 2.2.15

c) Spherical; Table 2.1 formula 2.2.17

d) Rational Quadratic; Table 2.1 formula 2.2.18

e) Wave Function; Table 2.1 formula 2.2.21.

f) Matérn Class of Functions; Table 2.1 formula 2.2.22, where $\theta_1=10;100; \theta_2=0.1$;

$\alpha=0.2$.

Another model which we will examine since it is related to our data is given in Armstrong (1998, p. 39), named prismato-magnetic model and described by:
\[ \gamma(h) = \begin{cases} \sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[1 - \frac{1}{(1 + (h/a)^2)^{1.5}}\right] & h > 0 \\ 0 & h = 0 \end{cases} \] (2.2.23)

Model fitting can be performed in many ways; Zimmerman & Zimmerman (1991) compared a few methods and concluded that minimizing the weighted sum of squares usually performs well and never does worse than other methods such as Maximum Likelihood. In practice, fitting a model to a sample variogram / covariance function / homeogram is a combination of visual and automatic methods. Pannatier (1996) used an entirely interactive visual method to fit a mathematical model to the sample variogram, but he developed an Indicator for the Goodness of Fit (IGF). The advantage of his program is that it measures the simultaneous fit of all nested structures, i.e. multiple directions are evaluated. Many other packages use automatic methods, which are iterative. This means that parameters, such as the ranges and the sills, are incrementally adjusted and statistically measured repeatedly until a solution model converges. Another approach to fit the variogram function is the maximum likelihood estimation (ML). Related to this class is a special method called the restricted maximum likelihood (REML), see Christensen (1993). REML reduces the problems associated with the regular maximum likelihood (ML) method which needs a well-defined Gaussian distribution in the sample data across all lags (Cressie, 1993). Minimization of errors between models is the goal, but nonstationarity makes it difficult to use ML well. We have examined a couple of geostatistical packages and only a few could handle data sets bigger than 1000 observations. Moreover, the iterative least-squares methods many times were unstable and converged to a local maximum, which was not the actual solution.
Algorithms based on maximum likelihood took very long time or would even cause computer system failure. None of the packages had homeogram function fitting or even covariance function fitting and, therefore, we had to develop our own program.

2.2.5 Experiments with fitting spatial coherency functions

The process of representing the spatial continuity mathematically is often long and requires previous knowledge and experience with the data source as Isaaks and Srivastava (1989) wrote: "The analysis of spatial continuity in a sample data set is often very frustrating, sometimes seemingly hopeless". Moreover, we should always be aware of the resolution of our data or random variables; at a different resolution the same process will have a different spatial dependence function. It is important to note that the most important feature of a spatial coherency function is its behavior near the origin; see Armstrong and Wackernagel (1988). A few steps are involved in the process of variogram / homeogram estimation as described in the following.

Plotting the sample coherency function:

Firstly, we checked and validated that the data in both of our case studies are isotropic. Next we used program Covar5.m (see Appendix A ) to read the X,Y,Z ASCII file of our test data and the inputted maximum distance parameter to produce Figures 2.3 or 2.4 as well as an output file with the lag values of semi-variogram, covariance function and homeogram (15 lags are presented in the plot -half of maximum distance- but all the 30 lags are in the output file). Program Covar5.m runs in time of $O(n^2)$ since for every point we have to compute the coherency with all the other point $(n(n-1)/2)$. An improvement for the computational time can be achieved by spatial sorting (see Section 5.2) and using
the maximum distance as a criterion to take only points within the limited distance neighborhood. Figure 2.3 and Figure 2.4 present the semi-variogram, homeogram and covariance function of the magnetic test data. In Figure 2.3 we did not limit the maximum pair distance; the resulting graph, which is not stationary, resembles a power function that may suggest a trend contamination. In the case of the magnetic data this may be explained by remainders of the Earth's magnetic field, which was removed using the IGRF earth magnetic field model (British Geological Survey 2001). When we limited the maximum distance of the pair selected to 15,000m, we got a more reasonable sample graph. In order to understand the data thoroughly we computed the 3rd order polynomial surface of our data in Figure 2.5. For the data with 1000 meter sampling interval (Magdata1.txt) the residuals variogram had a nugget effect model (flat line) i.e. there is no local variability or correlation between neighboring observations. For the data with 500 meters sampling interval (Magdata2.txt) the resultant variogram is given in Figure 2.6.

Figures 2.7 and 2.8 present the spatial coherency function for the laser scanning test data. These data as described in Section 1.3 have indeed a stationary behavior. However, for reasons presented in Chapter 4,5 we had to reduce the mean effect. The resultant graph, after performing detrending by means of a 3rd order polynomial as presented in Section 2.1.3, is given in Figure 2.8.
Figure 2.3: Sample semi-variogram, covariance function, homeogram of the test magnetic data when taking all the pairs including distant observations.

Figure 2.4: Sample semi-variogram, covariance function, homeogram of the test magnetic data when limiting observation separation distance to smaller then 15000m.
Figure 2.5: Trend analysis of the magnetic data, the grid was calculated using a 3rd order polynomial, with 130m grid resolution, and RMS error of 94.28m.

Figure 2.6: Semi-variogram, covariance function, homeogram of the residuals of the magnetic data after subtraction of the trend surface.
Figure 2.7: Sample semi-variogram, homeogram, covariance function of the laser scanning test data.

Figure 2.8: Semi-variogram, homeogram, covariance function of the laser scanning test data after detrending.
Fitting a model function: For the sample semi-variogram we used Surfer(2000) to fit a model function. Homeogram and covariance function fitting was not available in any of the programs that we checked, and we had to develop our own program for that. We used the method that was proposed by Chen and Jiao (2001), who transformed the theoretical functions of the different geostatistical models into corresponding linear ones. We use the Taylor expansion (to the $11^{th}$ element) of the function $e^{(h/r)}$ to linearize the exponential and Gaussian models as follows:

$$e^{-(h/r)} = 1 - \frac{h}{r} + \frac{h^2}{2!r^2} - \frac{h^3}{3!r^3} + ..$$  \hspace{1cm} (2.2.24)

so that the exponential model is:

$$C(h) = c_n + \sigma^2_y \left[ \exp\left(-\left(\frac{h}{r}\right)\right) \right] = c_n + \sigma^2_y \{1 - \frac{h}{r} + \frac{h^2}{2!r^2} - \frac{h^3}{3!r^3} + .. \frac{h^{11}}{11!r^{11}} \}$$

Let

$$\xi_0 = c_n + \sigma^2_y \hspace{1cm} \xi_1 = -\frac{\sigma^2_y}{r} \hspace{1cm} \xi_2 = \frac{\sigma^2_y}{2!r^2} \hspace{1cm} .. \hspace{1cm} \xi_{11} = -\frac{\sigma^2_y}{11!r^{11}}$$  \hspace{1cm} (2.2.25)

and for $h=h_1$ let

$$y_1 = C(h_1), \hspace{1cm} a_1(h_1) = h_1, \hspace{1cm} a_2(h_1) = h_1^2, \hspace{1cm} .. \hspace{1cm} a_{11}(h_1) = h_1^{11}$$

Then we can write for arbitrary $h_i$ ( $i=1, .., n$):

$$y_i = \xi_0 + \xi_1 a_1(h_i) + \xi_2 a_2(h_i) + .. + \xi_{11} a_{11}(h_i)$$

and

42
\[
\mathbf{A} = \begin{bmatrix}
1 & a_1(h_1) & a_2(h_1) & a_3(h_1) & \cdots & a_{11}(h_1) \\
1 & a_1(h_2) & a_2(h_2) & a_3(h_2) & \cdots & a_{11}(h_2) \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
1 & a_1(h_n) & a_2(h_n) & a_3(h_n) & \cdots & a_{11}(h_n)
\end{bmatrix}
\]  

\[ P_n - \] \text{is a diagonal weight matrix constructed by using the number of pairs in each lag (Cressie, 1993 p. 95). The computed parameters are:}

\[
\hat{\xi} = \begin{bmatrix}
\xi_0 \\
\xi_1 \\
\xi_2 \\
\vdots \\
\xi_{11}
\end{bmatrix} = \left(A^T \cdot P_n \cdot A\right)^{-1} \cdot A^T \cdot P_n \cdot \begin{bmatrix}
y_0 \\
y_1 \\
y_2 \\
\vdots \\
y_n
\end{bmatrix}
\]

(2.2.26.2)

from which we derive

\[ r = \frac{\xi_1}{(-2 \cdot \xi_2)}; \quad \sigma_y^2 = \xi_1 \cdot r; \quad c_{ii} = \xi_0 - \sigma_y^2. \]  

(2.2.27)

This fitting process of the exponential function gives us also the variance component and dispersion matrix for the estimated parameters following the standard adjustment computation process:

residual vector: \( \hat{\varepsilon} = y - A\hat{\xi} \)

estimated variance component \( \hat{\sigma}_o^2 = \frac{\hat{\varepsilon}^T P\hat{\varepsilon}}{n - m} \)  

(2.2.28)

estimated dispersion matrix:

\[ \hat{D}(\hat{\xi}) = \hat{\sigma}_o^2 N^{-1} \]

for \( N = A^T \cdot P_n \cdot A \)
We applied the same process to the Gaussian model, (2.2.15), to get the following equation:

\[ C(h) = c_{h} + \sigma_{y}^{2} \left[ \exp\left( -\frac{h}{r} \right) \right] = c_{h} + \sigma_{y}^{2} \left\{ 1 - \left( \frac{h}{r} \right)^{2} + \left( \frac{h^{4}}{2!\cdot r^{4}} \right) - \frac{h^{6}}{3!\cdot r^{6}} + \cdots + \frac{h^{22}}{11!\cdot r^{22}} \right\} \]

(2.2.29)

and the estimations:

\[ r = \sqrt{-\xi_{1}/(2 \cdot \xi_{1})}; \quad \sigma_{y}^{2} = -\xi_{1} \cdot r^{2}; \quad c_{h} = \xi_{0} - \sigma_{y}^{2}. \]

With the Gaussian model we got a better fit to the homeogram of the detrended laser data.

**Fitting a variogram model to the magnetic test data.**

We experimented with the models given in Table 2.1; the best fit was acquired using a nested Gaussian function (2.2.15) with the following parameters:

![Figure 2.9: Left: comparison of the exponential, Gaussian, and prismato-magnetic models. Right: zooming to the origin to detect micro-scale variation.](image-url)
Nugget: $\sigma_e^2 = 28.89$; Range (length): $r = 1360$ m; Sill (Scale) $\sigma_y^2 = 3752$ m.

Range (length) $r = 2087$ m; Sill (Scale) $\sigma_y^2 = 6238$ m.

which, fitted to the model, will give us (see also Figure 2.9):

$$
\gamma(h) = \begin{cases} 
28.8 + (3752 - 28.8) \left[ 1 - \exp\left( -\left( \frac{h}{1360} \right)^2 \right) \right] + (6238 - 28.8) \left[ 1 - \exp\left( -\left( \frac{h}{2087} \right)^2 \right) \right] & h > 0 \\
0 & h = 0
\end{cases}
$$

Experiments with the detrended magnetic data gave us the following variogram function:

Nugget: $\sigma_e^2 = 32.9$; Range (length): $r = 902.1$ m; Sill (Scale) $\sigma_y^2 = 5350$ m.

Range (length) $r = 2190$ m; Sill (Scale) $\sigma_y^2 = 3200$ m.

Fitting a homeogram and variogram to the laser scanning test data.

This was performed using the same programs described above (Surfer (2000) and fitfun.m). The results are given in Figure 2.10 and Table 2.2 with the original topographic data and the detrended topographic data.

<table>
<thead>
<tr>
<th>Semi-variogram</th>
<th>Homeogram</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Original test data</strong></td>
<td>$\eta(h) = \sigma_e^2 + \sigma_y^2 \left[ \exp\left( -\left( \frac{h}{r} \right) \right) \right]$</td>
</tr>
<tr>
<td>$\gamma(h)=0$</td>
<td>exponential model; $r = 62.96$, $\sigma_y^2 = 34.725$, $\sigma_e^2 = 1261.75$</td>
</tr>
<tr>
<td>$\gamma(h) = \sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[ 1 - \exp(-h/r) \right]$</td>
<td></td>
</tr>
<tr>
<td>$r=156$ m, $\sigma_e^2 = 0.468$, $\sigma_y^2 = 0.937$</td>
<td></td>
</tr>
</tbody>
</table>

| **Detrended data** | $\eta(h) = \sigma_e^2 + \sigma_y^2 \left[ \exp\left( -\left( \frac{h}{r} \right) \right) \right]$ |
| $\gamma(h)=0$ | exponential model; $r = 30.00$, $\sigma_y^2 = 0.45$, $\sigma_e^2 = 0.02$ |
| $\gamma(h) = \sigma_e^2 + (\sigma_y^2 - \sigma_e^2) \left[ 1 - \exp(-h/r) \right]$ | |
| $r=18.8$ m, $\sigma_e^2 = 0.046$, $\sigma_y^2 = 0.607$ | |

Table 2.2: Results of model fitting to the laser scanning test data.
Figure 2.10: Model fitting to the laser scanning test data. Upper figure is for the original data lower one for the detrended data. Solid line is the original sample homeogram function, dashed blue line is the LESS fitting and dashed red is the final function after interactive fine tuning of the homeogram function.
2.3 The geostatistical paradigm

Our main goal in this section is to review the formulas of an optimal linear predictor for the observed process, where optimality is obtained by minimizing the MSPE.

Mathematical Model

Our observed process is given by:

\[
\begin{align*}
y &= x + e \\
\mathbf{y} &= \mathbf{x} + \mathbf{e}_0
\end{align*}
\]

and

\[
\begin{bmatrix}
e \\ e_0
\end{bmatrix} \sim \begin{bmatrix}
\sigma_e^2 & I & 0 \\ 0 & 0 & C_x
\end{bmatrix}
\]

(2.3.1)

where

- \(n_y := [y(s_1),..., y(s_n)]^T\) is the \(n \times 1\) vector of observed process values
- \(n_x := [x(s_1),..., x(s_n)]^T\) is the \(n \times 1\) random effects vector of actual process value (our prediction is for \(x\) at location \(s'\))
- \(e := [e_1, ..., e_n]^T\) is the \(n \times 1\) random observation error
- \(e_0 := [e_01, ..., e_{0n}]^T\) is the \(n \times 1\) random process mis-centering vector
- \(\beta_i\) is the process mean value (usually unknown)
- \(nC_{xx} \) is the \(n \times n\) dispersion matrix of \(x\) with the elements \(C_x(s_i, s_j)\)
- \(\sigma_e^2\) is the variance component for the observational noise
- \(\tau_i = [1, 1, ..., 1]^T\) is the \(n \times 1\) "summation vector"

InhomBLIP or Simple Kriging formulas:

Let the process mean value be known (\(\beta = \beta_0\)). We can deduce the mean of the process if, for example, there are repetitions of the same phenomenon. Using the notations of Schaffrin (1993, 1997, 2000a) we introduce:

\[
nK_n := (I - \sigma_e^2 + C_x)
\]

(2.3.2)
as the covariance matrix of the observed process $y(s_i)$ in the sample points, and

$$\kappa(s') := [C_x(s_i, s'), \ldots, C_x(s_n, s')]^T \tag{2.3.3}$$

as the corresponding vector of covariances between sample points $s_i$ and the new point $s'$.

Then the Simple Kriging or inhomBLUP is given by:

$$\bar{x}_s(s') = \beta_0 + \kappa(s')^T \cdot K^{-1} (y - \tau \beta_0) \tag{2.3.4}$$

with the Mean Square Prediction Error (MSPE)

$$MSPE[\bar{x}_s(s')] = \sigma_x^2 - \kappa(s')^T K^{-1} \kappa(s') \tag{2.3.5}$$

**HomBLUP predictor or Ordinary Kriging:**

In most real-world problems the mean is not known and we have to use the homBLUP or Ordinary Kriging predictor which also estimates the process expectation $\beta$. We will summarize the results to predict the process value at an unobserved point $s' \in \mathbb{R}^2$ (see Appendix B for a complete derivation, and also Schaffrin and Bock (1994)):

$$\bar{x}(s') = \{C_x(s', s_i)] - \tau \cdot \frac{[\sigma_x^2 \cdot 1 + C_x]^{-1} [C_x(s', s_i)]} {\tau \cdot [\sigma_x^2 \cdot 1 + C_x]^{-1} \cdot \tau} \cdot [\sigma_x^2 \cdot 1 + C_x]^{-1} \cdot y \tag{2.3.6}$$

The BLUE (or LESS) of the mean of both $y(s)$ and $x(s)$ is independent of $s$, namely:

$$\hat{\beta} = \frac{\tau \cdot K^{-1} \cdot y}{\tau \cdot K^{-1} \cdot \tau} \tag{2.3.7}$$

We plug (2.3.7) and (2.3.2-3) into (2.3.6) to shorten our formulas to

$$\bar{x}(s') = \hat{\beta} + \kappa(s')^T \cdot K^{-1} (y - \tau \cdot \hat{\beta}) \tag{2.3.8}$$

with the Mean Square Prediction Error:
\[
\text{MSPE}[\mathbf{x}(s')] = \sigma_x^2 - \kappa(s')^T \mathbf{K}^{-1} \kappa(s') + \frac{[1 - \kappa(s')^T \mathbf{K}^{-1} \cdot \tau]^2}{\tau^T \cdot \mathbf{K}^{-1} \cdot \tau} \tag{2.3.9}
\]

**Primal Vs Dual System:** The Kriging computations (2.3.4)-(2.3.9) can be performed in a dual system or primal system; we take for example the Ordinary Kriging case (homBLUP). From (3.2.5) we write the primal system (Schaffrin 1997):

\[
\begin{bmatrix}
K & -\tau \\
-\tau^T & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{x}(s) \\
\mathbf{v}(s)
\end{bmatrix}
= 
\begin{bmatrix}
\kappa(s) \\
-1
\end{bmatrix}
\]

or

\[
\begin{bmatrix}
\mathbf{x}(s) \\
\mathbf{v}(s)
\end{bmatrix}
= 
\begin{bmatrix}
K & -\tau \\
-\tau^T & 0
\end{bmatrix}^{-1}
\begin{bmatrix}
\kappa(s) \\
-1
\end{bmatrix}
\tag{2.3.11}
\]

Thus we get directly:

\[
\mathbf{x}(s) = \mathbf{x}(s)^T \cdot \mathbf{y}
\tag{2.3.12}
\]

and

\[
\text{MSPE}[\mathbf{x}(s)] = \sigma_x^2 + \mathbf{x}(s)^T \cdot \kappa(s) + \mathbf{v}(s)
\]

The dual system on the other hand is

\[
\begin{bmatrix}
K & \tau \\
\tau^T & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{g} \\
\hat{\mu}
\end{bmatrix}
= 
\begin{bmatrix}
\mathbf{y} \\
\mathbf{0}
\end{bmatrix}
\tag{2.3.13}
\]

or

\[
\begin{bmatrix}
\mathbf{g} \\
\hat{\mu}
\end{bmatrix}
= 
\begin{bmatrix}
K & \tau \\
\tau^T & 0
\end{bmatrix}^{-1}
\begin{bmatrix}
\mathbf{y} \\
\mathbf{0}
\end{bmatrix}
\]

Consequently we need to compute \( \mathbf{g} \) one time for our data and then apply:

\[
\mathbf{x}(s) = \hat{\mu} + \kappa(s)^T \cdot \mathbf{g}
\tag{2.3.14}
\]

with a different \( \kappa(s) \) vector at each point. However, the dual system does not give us a direct way to compute the mean square prediction error. Kriging equations in terms of the semi-varioagram function are derived in Cressie (1993, pp. 121-123) and Schaffrin (2001). We define
\[
\gamma(s') := [\gamma(s'-s_1), \ldots, \gamma(s'-s_n)]^T \tag{2.3.15}
\]

\[
\Gamma := [\gamma(s_i - s_j), \quad i, j \in \{1, \ldots, n\}.
\]

The primal kriging system is
\[
\begin{bmatrix}
\Gamma & \tau^T \\
\tau & 0
\end{bmatrix}
\begin{bmatrix}
\overline{x}(s') \\
\overline{v}(s')
\end{bmatrix} =
\begin{bmatrix}
\gamma(s') \\
1
\end{bmatrix} \tag{2.3.16}
\]

which yields
\[
\overline{x}(s') = \overline{x}(s')^T \cdot y \tag{2.3.17}
\]

and
\[
MSPE(\overline{x}(s')) = \overline{x}(s')^T \cdot \gamma(s') + \overline{v}(s') \tag{2.3.18}
\]

In the development of Delaunay structured kriging programs (Chapter 3 and 4) we shall use equations (2.3.16)-(2.3.18) since the primal system gives a direct way to compute the MSPE. The equations using the homeogram are:

\[
\eta(s') := [\eta(s'-s_1), \ldots, \eta(s'-s_n)]^T \tag{2.3.19}
\]

\[
H := [\eta(s_i - s_j), \quad i, j \in \{1, \ldots, n\}, \tag{2.3.20}
\]

\[
\begin{bmatrix}
H & -\tau \\
-\tau^T & 0
\end{bmatrix}
\begin{bmatrix}
\overline{x}(s') \\
\overline{v}(s')
\end{bmatrix} =
\begin{bmatrix}
\eta(s') \\
-1
\end{bmatrix} \tag{2.3.21}
\]

\[
\overline{x}(s') = \overline{x}(s')^T \cdot y \tag{2.3.22}
\]

\[
MSPE(\overline{x}(s')) = \eta(0) - \overline{x}(s')^T \cdot \eta(s') + \overline{v}(s') \tag{2.3.23}
\]

The last terms will be used in Chapter 5. The dual system for Ordinary Kriging along with equivalent systems for other forms of Kriging are given in Schaffrin (2001).
CHAPTER 3

SPATIAL DATA STRUCTURE IN SUPPORT FOR

GEOSTATISTICAL ALGORITHMS

This chapter reviews computational geometry algorithms and data structures that are used to select the neighborhood of geostatistical data processing. Two very principal structures, the Delaunay triangulation and the Voronoi diagram, will be described with their main properties. We summarize the computational complexity or the running time of the reviewed algorithms. Based on this review, we suggest and develop a recursive algorithm that uses the Delaunay triangulation data structure to define the neighborhood in which the geostatistical method is applied.

3.1 Search neighborhood of geostatistical algorithms

The problem of fast neighborhood search is an extensively researched topic in the computer science field; for example, the classical book by Samet (1990) describes many possibilities to perform proximity search. However, the goal of many computer science researchers is to find a faster algorithm that is able to handle massive datasets in a computationally efficient way. We will look at the search problem from both the statistical and computational point of view. There is a very strong relationship between the search neighborhood and the accuracy of our prediction; this relationship stems from the fact that defining a wrong neighborhood may lead to extrapolation while a good
neighborhood definition will take all the closest points that also surround the questionable point in all directions, i.e. the case of interpolation. Another related issue is the problem of clustering within the data; Ordinary Kriging, in contrast to methods like inverse distance weighting, does attempt to account for the possibility of clustering through the inverse covariance matrix, thereby nearly canceling the effect of clustered points on the interpolated point. However, as we will demonstrate (in Chapter 4), this does not solve the case with a severely irregular pattern. Consequently, by using also Watson (1992, pp. 69-73), we can write the following requirements for neighborhood selection (also called subset selection):

- The selected subset should be distributed uniformly around the interpolation point in a way that will avoid under-representation of one direction as well as extrapolation.
- The selection method should not allow excessively large or small subsets since large subsets tend to submerge the local details and small subsets will not give us enough information to predict the point accurately.
- Individual data points should not mask other data in the subset. This phenomenon, also called "screening effect", often occurs when nonzero Kriging weights are concentrated on a subset of samples in the immediate vicinity of the estimated point or block. Then this sample screens off the influence of all the other data.
- Using more samples increases the amount of computations required; for example, in Ordinary Kriging the number of computations necessary to solve for
the Ordinary Kriging weights is proportional to the cube \(O(n^3)\) of the number of points (Isaaks and Srivastava 1989, p. 341 and also Section 2 of Chapter 5).

- As data points come from farther and farther away, the appropriateness of a stationary random function model becomes more doubtful. Consequently Ordinary Kriging with local search neighborhoods allows one to account for local departures from stationarity over the area (Goovaerts 1997, pp. 178-179).

Using those requirements we will be able to design a better subset selection scheme. But before that we will look at the current search strategies.

3.1.1 Review of current search strategies

Search strategies for geostatistical algorithms are presented by several authors, among them: Isaaks and Srivastava (1989, pp. 338-350), Watson (1992, pp. 69-73), Goovaerts (1997, pp. 178-179), Deutsch and Journel (1997, pp. 32-37), and in program manuals of GIS software packages, for example: Surfer (2000), and ESRI (1995). However, a full discussion of the statistical and computational properties of each method, to our best knowledge, is still missing. We summarize the five main approaches for subset selection:

- **Fixed Number Selection** – The subset is selected by choosing the nearest fixed number of data points (Isaaks and Srivastava 1989, p. 341, suggest at least 12 points). The advantage of this method is that its algorithm is simple and can be optimized for inversion of a matrix with fixed size (the number of points). The disadvantage is obvious when the points are irregularly distributed and clustered in one place and we, thus, are forced to extrapolate.
• Fixed Distance Subsets - The neighborhood of points is selected on the basis of fixed distance by including every point that is within a circular region centered on the interpolation location. Again, this approach works well for data that are uniformly and regularly spaced. However, when the data are clustered in one place more than in another then this method tends to choose too many points in the dense area and too few in the sparse area. An elaboration of this method is to use the anisotropy in the pattern of spatial continuity and to define an ellipse centered on the point being estimated. The ellipse is oriented with its major axis parallel to the direction of maximum continuity. Another elaboration is the fixed area data partitioning method in which we maintain a fixed area criterion and we allow the selection to change shape and orientation. Naturally, it is very difficult to implement such an algorithm in an efficient way so that it works fast enough.

• A better neighborhood search approach is the Quadrant Search which has some good properties both in terms of speed and the equal representation of different directions. Quadrant search typically specifies a maximum number of sample points in any particular quadrant. If a particular quadrant has fewer sample points than the maximum allowable, then we keep all the points it contains; however, if a quadrant contains too many sample points then we keep only the closest ones. The quadrant search can be implemented using Quadtree, a hierarchical data structure method with a very efficient search neighborhood algorithm. A particular algorithm which is used in IDRISI (2000) is the PMR Quadtree described in Hjaltason and Samet (1995). This algorithm is based on the principle
of recursive decomposition of the space into regular squares, each having a specified number of points. There are many more hierarchical data structures that can be used in a neighborhood search; they are reviewed in Samet (1995), for example.

- A more complex approach is the Octant Search approach which divides the map area into eight sectors around the required point, and finds the closest data points in each of the eight sectors during interpolation. This method is described by Armstrong (1998, pp.123-124) as the angular sectors method.

- Another variant of the quadrant search is to use a Square Search neighborhood and then to look for a composite sample (i.e. a condensation of many points into a single point where the average covariance to all these points is assigned as the covariance to the composite point) in the quadrant outside the square; for more details, see (Isaaks and Srivastave 1989, p. 342).

Figure 3.1 portrays the common search methods: circle search neighborhood, ellipse and quadrant. Note that only the ellipse method gives a representation of all directions in this particular case.
3.2 Delaunay triangulation data structure

3.2.1 Introduction to triangulation

According to Lloyd (1977) we can formulate triangulation as follows: Let $P$ be a set of $n$ distinct points in the plane. The points in $P$ are called vertices. Let $L$ be the set of all straight line segments between any pairs of vertices in $P$. The elements of $L$ are called edges. Two edges, $e$ and $f$, are said to properly intersect if they intersect at a point which is not an endpoint of each, and $e \neq f$. A triangulation of $P$ is a maximal subset $T$ of $L$ such that no two edges of $T$ properly intersect. This implies that each face of the straight-line planar graph determined by $P$ and $T$ is a triangle. Following this definition we notice that a triangulation can have one of many properties, such as the minimum weighted distances triangulation – MWT, or the "greedy" triangulation and, of course, the Delaunay triangulation.
Number of edges and triangles in a triangulation – According to de Berg et al. (1997), let $P$ be a set of $n$ points in the plane, not all collinear, and let $k$ denote the number of points in $P$ that lie on the boundary of the convex hull of $P$. Then for any triangulation of $P$ we have

$$\text{Number of triangles} = 2n-2-k,$$

$$\text{Number of edges} = 3n-3-k.$$ 

(3.1.1)

The proof for those relationships is derived from Euler’s formula.

3.2.2 Properties of the Delaunay triangulation:

The Delaunay triangulation $D(P)$, named after Boris Delaunay for his pioneering work in Delaunay (1934), is a triangulation in the sense of Lloyd which is characterized by the fact that the circumcircle of each triangle contains no other points of the set $P$. In order to check this property we use the in-circle-test formula, given in Cartesian coordinates by:

$$\det \begin{vmatrix}
X^2 + Y^2 & X & Y & 1 \\
X_1^2 + Y_1^2 & X_1 & Y_1 & 1 \\
X_2^2 + Y_2^2 & X_2 & Y_2 & 1 \\
X_3^2 + Y_3^2 & X_3 & Y_3 & 1 \\
\end{vmatrix} > 0 \quad \text{iff } P \text{ is inside the circle passing through } P_1, P_2 \text{ and } P_3$$

(3.1.2)

where $X_1, Y_1, X_2, Y_2, X_3, Y_3$ are the Cartesian coordinates of the three points in clockwise order, and $X, Y$ is the new point that should not be inside the circle in order for the triangle $\Delta P_1P_2P_3$ to be part of the Delaunay triangulation (for a proof of the circle test formula see Guibas and Stolfi 1985). If the above determinant is equal to 0 then the four points are co-circular and we have ambiguity.
Figure 3.2: The point P does not fulfill the in-circle test; however, if the point P is replaced by P’, then we have ambiguity in the determination of the Delaunay triangulation.

Other unique properties of the Delaunay triangulation according to de Berg et al. (1997), or Saalfeld (1998b) include:

1. The disk property – If no four points are co-circular (i.e. lie on the same circle) then an edge will belong to the (unique) Delaunay triangulation if and only if there exists a closed disk containing both endpoints of the edge and no other point of the vertex set.

2. Maximizing the minimum angles – This property is the chief merit of the Delaunay triangulation from the interpolation point of view, namely among all possible triangulations it maximizes the minimum interior angle of the whole triangulation. In fact, a much stronger statement holds as well: Among all triangulations with the same smallest angle, the Delaunay triangulation maximizes the second smallest angle, and so on. In particular, any triangulation can be associated with a sorted angle sequence, that is,
the increasing sequence of angles \((\alpha_1, \alpha_2, \ldots, \alpha_m)\) appearing in the triangles of the triangulation.

3. An interesting property of Delaunay triangulation (that does not hold for arbitrary triangulations) is the possibility of a partial ordering with respect to any viewpoint in the plane; see Nagy (2000). In other words, for a given viewpoint, the Delaunay triangles can be assigned consecutive numbers as labels so that from any arbitrary point in any triangle we can draw a line to this viewpoint that passes only through triangles with decreasing number. Naturally, the partial ordering depends on the chosen viewpoint.

Figure 3.3: A partial ordering of Delaunay triangles; any ray to the viewpoint from an arbitrary point will pass only through triangles with decreasing numbers.

There are many more interesting properties which can be found in computational geometry books such as Aurenhammer and Klein (2000, pp. 225-229) or O'Rourke (1998, pp. 161-163). A disadvantage of the Delaunay triangulation for arranging data is that, whenever four points are co-circular and the angles have the same order (e.g. in a
square), we have ambiguity and there are two options for the triangulation; see Figure 3.4. This is due to the lack of suitable information in our data. Another disadvantage is that the Delaunay triangulation does not necessarily have the closest point in its zero-order triangulation. We present a counter-example where a point \( P \) within a triangle and its closest point is only in the 4\(^{th} \) order triangulation (see the definition of triangulation order in section 3.2.4).

![Figure 3.4: The ambiguity or non-uniqueness in the determination of the Delaunay triangulation; after projecting the 3-D situation to the plane the decisive information may have been lost (adapted from Saalfeld 98).](image)

Counter-example information: The point \( P \) has the plane coordinates of 220,270 while the other point coordinates are (all the coordinates are in meters):

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>180.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>130.00</td>
<td>280.00</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>280.00</td>
<td>280.00</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>280.00</td>
<td>300.00</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>220.00</td>
<td>310.00</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>170.00</td>
<td>300.00</td>
<td></td>
</tr>
</tbody>
</table>

Thus we get the following ordered Euclidean distances (in meters) between the points:

\[
\begin{align*}
\text{d}(P,5) &= 40.00 \\
\text{d}(P,6) &= 58.30 \\
\text{d}(P,3) &= 60.82 \\
\text{d}(P,2) &= 90.55 \\
\text{d}(P,4) &= 67.08 \\
\text{d}(P,1) &= 253.17
\end{align*}
\]
Figure 3.5: Counter-example for the case of the closest point to P not belonging to the
triangle \( \triangle 123 \). Lines between points are from Delaunay triangulation, and polygons are
Voronoi diagram polygons.

3.2.3 Using TIN for piecewise linear interpolation

We want to compute the value \( Z_P \) at point \( P \) inside a triangle, using the three surrounding
points \( P_1, P_2, P_3 \) with the values \( Z_1, Z_2, Z_3 \) respectively. We express the Cartesian
coordinate of \( P(X_P, Y_P) \) uniquely as weighted average by:

\[
X_P = \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3 \\
Y_P = \alpha_1 Y_1 + \alpha_2 Y_2 + \alpha_3 Y_3
\]

where \( \alpha_1 + \alpha_2 + \alpha_3 = 1 \), and the resulting computed value is:

\[
Z_P = \alpha_1 Z_1 + \alpha_2 Z_2 + \alpha_3 Z_3 \tag{3.2}
\]
The unique weights $\alpha_1, \alpha_2, \alpha_3$ are called the convex coordinates (or barycentric coordinates) of $P$ and can be computed using various methods; one of them is provided by Saalfeld (1998b) and Cressie (1993, pp. 373-374):

$$\alpha_1 = \frac{\text{Area}(\Delta PP_2 P_3)}{\text{Area}(\Delta P_1 P_2 P_3)},$$

$$\alpha_2 = \frac{\text{Area}(\Delta P_1 PP_3)}{\text{Area}(\Delta P_1 P_2 P_3)},$$

$$\alpha_3 = \frac{\text{Area}(\Delta P_1 P_2 P)}{\text{Area}(\Delta P_1 P_2 P_3)}.$$  \hfill (3.3)

The area of the triangle can be computed from the length of the edges using Heron's formula: \[ \text{Area} = \sqrt{s(s-a)(s-b)(s-c)} \] when \( s = (a+b+c)/2 \) \hfill (3.4)

and \( a, b, c \) denote the edges of the respective triangle. By rewriting equation (3.2) we use another form to compute the value of a point inside the triangle, as shown in Figure 3.6:

![Figure 3.6: Linear interpolation in a triangle adapted from Terei (1999).](image)

From (3.2) and (3.3) we obtain:

$$-Z_p \cdot \text{Area}(\Delta P_1 P_2 P_3)_{\text{proj}} + Z_{p_1} \cdot \text{Area}(\Delta PP_2 P_3)_{\text{proj}} + Z_{p_2} \cdot \text{Area}(\Delta P_1 PP_3)_{\text{proj}}$$

$$+ Z_{p_3} \cdot \text{Area}(\Delta P_1 P_2 P)_{\text{proj}} = 0,$$

and with the matrix representation of the area of a triangle
We get another form of equation (3.2), namely:

\[
Z_P \cdot \begin{vmatrix} X_1 & Y_1 & 1 \\ X_2 & Y_2 & 1 \\ X_3 & Y_3 & 1 \end{vmatrix} - Z_1 \cdot \begin{vmatrix} X_p & Y_p & 1 \\ X_2 & Y_2 & 1 \\ X_3 & Y_3 & 1 \end{vmatrix} + Z_2 \cdot \begin{vmatrix} X_p & Y_p & 1 \\ X_1 & Y_1 & 1 \\ X_3 & Y_3 & 1 \end{vmatrix} - Z_3 \cdot \begin{vmatrix} X_p & Y_p & 1 \\ X_1 & Y_1 & 1 \\ X_2 & Y_2 & 1 \end{vmatrix} = 0
\]

or

\[
\begin{vmatrix} X_p & Y_p & 1 & Z_P \\ X_1 & Y_1 & 1 & Z_1 \\ X_2 & Y_2 & 1 & Z_2 \\ X_3 & Y_3 & 1 & Z_3 \end{vmatrix} = 0
\]

(3.5.3)

Coplanarity of point \( P \) with \( P_1, P_2, \) and \( P_3 \) is satisfied by the vanishing determinant of (3.5.3); see also Terei (1999). Triangular Irregular Network (TIN) based linear interpolation has the advantages of being fast, easy to control and to understand, with equal representation of different directions within each triangle. For those reasons it is often used to interpolate relatively dense data sets of elevation points to form digital Elevation Models (DEM). However, this simplicity of TIN interpolation has many disadvantages:

- It does not enable us to improve and adjust our observations,
- It does not give us a good measure of the mean squared error (such as the MSPE) since the mostly used error formula is based on the assumption of flat surfaces.

\[
\sigma_{z_P} = \sigma_{z_0} \cdot \sqrt{\alpha_1^2 + \alpha_2^2 + \alpha_3^2}
\]

(3.5.4)
• It is not quite appropriate for smooth surfaces like equi-potential surfaces or environmental values, since its first derivative does not exist along the edges.

• It is less robust against outliers since the interpolation uses no more than three points, no matter what.

• It is ambiguous in the case of four co-circular points when the determinant in (3.1) is equal to 0.

• It does not take into account the type of surface, its smoothness properties (or spatial variability) in particular.

3.2.4 Defining the order of Delaunay neighborhood

In view of the advantageous properties of Delaunay triangulation we can subdivide our dataset in a way that will represent different directions in the best way (due to the maximization-of-the-minimum angles property). Thus it will be very reasonable to use this Delaunay data structure for the Kriging prediction; but in order to do that we will have to develop some basic terminology. We develop a definition for the interpolation neighborhood of irregularly distributed point-sets; this definition is using the edges to expand the neighborhood in contrast to using the triangle vertices to define participating triangles.

Zero-order Delaunay neighborhoods will have exactly the three points (vertices) which make up the triangle surrounding the point at which we want to predict its value.

First-order Delaunay neighborhoods will have four to six points: three from the zero-order triangulation, and one to three vertices from the triangles that have a common edge with the previously defined zero-order triangle. For example: when the zero-order
neighborhood includes a triangle whose three vertices lie on the convex hall, only four vertices will be in the first-order neighborhood, namely, three of zero-order and one of first-order.

**Second-order Delaunay neighborhoods** will have a minimum of five points and a maximum of 12 points (see also Appendix C). Following the same formalism, we can recursively define the **n\textsuperscript{th}-order Delaunay neighborhoods** as: all the points, which are vertices of triangles that share an edge with the triangulation constructed from the (n-1)-order neighborhood of points. In Figure 3.7 we take a triangulation and demonstrate the concept of orders of triangulations. Now, we can define "n\textsuperscript{th}-order TIN structure Kriging" to be Kriging which uses the data points from the n\textsuperscript{th}-order Delaunay triangulation neighborhood.

![Figure 3.7](image)

**Figure 3.7:** [A] The initial triangulation with the red square as the location at which we want to predict its value. [B] The second order Delaunay neighborhood. [C] The first order Delaunay neighborhood.

An interesting empirical investigation was performed by Midtbø (1993) who studied the geometrical influence region of a new point in the Delaunay triangulation. The
experiments were conducted by constructing an initial triangular network, with an incremental algorithm adding the new node and checking the number of edge swaps and the level on which the new points had an influence. 15 datasets of 1000 randomly distributed points were used for the experiments. The results are given in Table 3.1.

<table>
<thead>
<tr>
<th>average number of edge swaps</th>
<th>3009</th>
</tr>
</thead>
<tbody>
<tr>
<td>highest level of edge swaps</td>
<td>6</td>
</tr>
<tr>
<td>average max edge swaps per point</td>
<td>9.3</td>
</tr>
<tr>
<td>Edge swaps at various levels (%)</td>
<td></td>
</tr>
<tr>
<td>Level 0</td>
<td>63.0</td>
</tr>
<tr>
<td>Level 1</td>
<td>28.7</td>
</tr>
<tr>
<td>Level 2</td>
<td>6.8</td>
</tr>
<tr>
<td>Level 3</td>
<td>1.8</td>
</tr>
<tr>
<td>Level 4</td>
<td>0.3</td>
</tr>
<tr>
<td>Level 5</td>
<td>0.03</td>
</tr>
</tbody>
</table>

Table 3.1: Geometric region of influence from Midtbø (1993).

A note on distance and topology relationship.

We would like to get an estimate on the relationship between the number of triangles included in the interpolation and the average maximum distance of the points from the center of the triangle. We cannot get an exact measure for that relationship since the points are in an irregular pattern or location; we know, however, that Delaunay triangulation maximizes the minimum angle (see Section 3.2.2, property 2) and this property attempt to make the triangulation as a uniform as possible. Consequently, in order to gain a better understanding on the topology and distance relationship we take two examples, the equilateral triangle mesh and the regular grid (see Figure 3.8) and try
to compute this relationship for those shapes. We start with computing the average triangle edge distance by using the area of triangle equation:

\[ \text{Area} = \text{edge}^2 \cdot \frac{\sqrt{3}}{4} \]

In the case of the equilateral triangular mesh the length of the edge is:

\[ \text{edge} = \frac{4 \cdot \text{Area}}{\sqrt{3} \cdot n_t} \]  \hspace{1cm} (3.6.1)

where \( n_t \) is the number of triangles and Area is the total area of the project's convex hull.

Using Pythagoras theorem we compute the distance between the center of gravity to the farthest point in the first-order triangulation \( d_1 \) (see Figure 3.8) which is:

\[ d_1 = \text{edge} \cdot \frac{3}{4} + \frac{1}{3} \cdot \text{edge} \cdot \frac{3}{4} = \text{edge} \cdot \frac{2\sqrt{3}}{3} \]  \hspace{1cm} (3.6.2)

and continue with

\[ d_2 = \sqrt{d_1^2 + \text{edge}^2} = \text{edge} \cdot \frac{\sqrt{21}}{3} \]  \hspace{1cm} (3.6.3)

At this point the only unknown variable is the number of triangles \( n_t \), which we can compute either by using (3.1.1) in the worst case scenario.

\[ n_t = 2n-2-k \]

Or for the case of equilateral mesh we can compute the exact number by observing the relationship between the number of levels \( L \) (see Figure 3.8) and the number of triangles. We use a geometric series to induce the following relationships:

\[ n_t = L^2 \]

\[ k=3 \cdot L \]
\[ n-k=(L-1)(L-2)/2 \]  

\[ n = 3 \cdot L + (L - 1)(L - 2)/2 = (L + 1) \cdot (L + 2)/2 \]  

\[ L = -\frac{3}{2} \pm \frac{\sqrt{1+8 \cdot n}}{2} \approx \frac{\text{for large } n}{\sqrt{2n}} \]  

\[ n_r = L^2 = \left( -\frac{3}{2} \pm \frac{\sqrt{1+8 \cdot n}}{2} \right)^2 \]  

\[ \text{edge} = \sqrt{\frac{4 \cdot \text{Area}}{\sqrt{3}}} / \left( -\frac{3}{2} \pm \frac{\sqrt{1+8 \cdot n}}{2} \right) \approx \frac{2 \cdot \text{Area}}{\sqrt{3} \cdot n} \text{ for large } n \]  

\[ d_1 \approx \sqrt{\frac{2 \cdot \text{Area}}{\sqrt{3} \cdot n}} \cdot \frac{2\sqrt{3}}{3} = \frac{2}{3} \cdot \sqrt{\frac{2 \cdot \text{Area} \cdot \sqrt{3}}{n}} \]  

Figure 3.8: Equilateral triangle and square meshes.
The same geometric derivation can be performed also on a square triangular mesh and we summerize the results in table 3.2

<table>
<thead>
<tr>
<th>Mesh Shape</th>
<th>No. of levels</th>
<th>Edge of the triangles</th>
<th>No. of triangles $n_t$</th>
<th>No. of convex hull points $n_{ch}$</th>
<th>No. of internal points $n_{int}$</th>
<th>Max. distance of 1$^{st}$ order triangle $\text{Max. distance of 2$^{nd}$ order triangle}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Triangle</td>
<td>$L$</td>
<td>edge</td>
<td>$L^2$</td>
<td>$3.5L$</td>
<td>$(L-1),(L-2)/2$</td>
<td>$\text{edge \cdot } \sqrt{3}/3$</td>
</tr>
<tr>
<td>Square</td>
<td>$L$</td>
<td>edge, edge $\cdot \sqrt{2}$</td>
<td>$2L^2$</td>
<td>$4L$</td>
<td>$(L-1)^2$</td>
<td>$1.581. \text{edge}$</td>
</tr>
</tbody>
</table>

Table 3.2: Properties of the equilateral triangle and square mesh.

We have examined some measures to evaluate the distance of the first-order neighborhood and thus we are able to compare between the semivariogram range (distance) and the level of triangulation (topology) that we use in the interpolation program. One simple method, that is used in practice, is an interactive evaluation of this distance using a GIS and checking the distance on a few characteristic triangles.

### 3.3 Voronoi diagram

The Voronoi diagram is named after the mathematician M. G. Voronoi who explored this geometric construction in 1908. It is being used in many different fields of science with different names, e.g. Thiessen polygons after the climatologist A. H. Thiessen, Dirichlet tessellation after G. L Dirichlet, Wigner-Seitz cells or Blum's transform in biology. A Voronoi diagram records information about what is close to what.

Let $P = \{ P_1 ; P_2 ; \ldots ; P_n \}$ be a set of points in the plane (or in any higher dimensional space), which we call sites. Define $V(P_i)$, the Voronoi cell for $P_i$, to be the open set of
points \( Q \) in the plane such that \( \text{dist}(Q; P_i) < \text{dist}(Q; P_j) \) for all \( j \neq i \). That is, the Voronoi cell for \( P_i \) consists of the set of points \( Q \) for which \( P_i \) is the unique nearest neighbor:

\[
V(P_i) = \{ Q \mid \text{dist}(P_i; Q) \leq \text{dist}(P_j; Q), \forall j \neq i \} \tag{3.7.1}
\]

Another way to define \( V(P_i) \) is in terms of the intersection of halfplanes. Given two sites \( P_i \) and \( P_j \), the set of points that are strictly closer to \( P_i \) than to \( P_j \) is just the open halfplane whose bounding line is the perpendicular bisector between \( P_i \) and \( P_j \) (see ORourke 1998, pp.158-159). The boundaries between Voronoi cells form the Voronoi diagram (see Figure 3.9).

![Figure 3.9: Voronoi diagram of a set of points.](image)

**3.3.1 Properties of the Voronoi diagram**

We will summarize a few properties, which are pertinent for our research:

**General:** The number of vertices in the Voronoi diagram for a set of \( n \) point sites in the plane is at most \( 2n - 5 \) and the number of edges which are connected to vertices is at most \( 3n-6 \), see proof by de Berg et al. (1997, p. 148). The average number of edges in the boundary of a Voronoi region/cell is less than 6. (Aurenhammer and Klein 2000, p. 207)
Voronoi edges: Each point on an edge of the Voronoi diagram is equidistant from its two nearest neighbors \( P_i \) and \( P_j \). Thus, there exists a circle centered at such a point so that \( P_i \) and \( P_j \) lie on this circle, and no other site is inside the circle.

Voronoi vertices: It follows that a vertex at which three Voronoi cells \( V(P_i), V(P_j), \) and \( V(P_k) \) intersect is equidistant from all three sites. Thus it is the center of the circle passing through these sites, and this circle contains no other sites in its interior.

Voronoi regions: Each Voronoi region \( V(P) \) is convex.

Duality: This is a persistent theme in computational geometry, and it is to transform one problem into a companion problem (like the Fourier transform in data analysis which transforms a spatial domain problem into the frequency domain). For example, the set of problems involving points can be transformed into problems involving lines by a method of duality, and in general \( T^{-1}(T(P)) = P \), where \( T^{-1} \) denotes the inverse transformation.

Delaunay triangulation is a straight-line dual of the Voronoi diagram; namely, each node of \( D(P) \) corresponds to an edge of \( V(P) \), each face (triangle) of \( D(P) \) corresponds to a vertex of \( V(P) \) and so on. The ability of the Voronoi diagram to record the proximity information of a set of points is very important for interpolation purposes since we have already seen that in kriging the closest point will have higher weight and influence on the predicted value. A Voronoi diagram's spatial structure does promise that the closest point will indeed be in the neighborhood. However, it does not guarantee equal representation of the different directions surrounding the point (extrapolation). In Figure 3.10 the center triangular point will be interpolated using only points to its right (25, 23, 10, 24, 31, 8, 12) which belong to neighboring cells.
Figure 3.10: Voronoi diagram neighborhood may cause extrapolation inadvertently.

3.4 Computational complexity of neighborhood search algorithms

The complexity of an algorithm is identified by its time and space requirements. Time is measured by the number of operations executed, and computer space by the number of variables needed to be stored at one time (for input, intermediate results and output). Complexity notation can be useful in summarizing an algorithm's expected performance, either as based on theoretical analysis or as measured from well described tests. The theoretical analysis of complexity is done by highlighting which components are dependent on the number of points and therefore play an important part in the algorithm. We use the $O$ notation of complexity to denote the general behavior of an algorithm as a function of the size of our input data:
Given any function $g(n)$, $O(g(n))$ is the set of functions, $f$, such that

$$0 \leq f(n) \leq cg(n) \text{ for all } n \geq n_0$$

where $c$ and $n_0$ are positive constants.

We say that $O(g(n))$ is the set of all functions which grow no faster than $g(n)$. A multiple of the function $g(n)$ is an upper bound to the functions in $O(g(n))$ for large $n$. In the analysis of our algorithms we will also use the following definitions:

**Worst-case time:** It is the maximum running time over all (legal) inputs of size $n$. Let $I$ denote a legal input instance, and let $|I|$ denote its length, and let $T(I)$ denote the running time of the algorithm on input $I$, then:

$$T_{\text{worst}}(n) = \max_{|I|=n} T(I)$$

**Average-case time:** It is the average running time over all legal inputs of size $n$. More generally, for each input $I$, let $p(I)$ denote the probability of seeing this input. The average-case running time is the weighted sum of running times, with the probability being the weight, usually:

$$T_{\text{avg}}(n) = \sum_{|I|=n} p(I)T(I)$$

The following definitions are from Preparata and Shamos (1985, p. 37):

**Query time:** It is the time that is required in both average and worst cases to respond to a single query.

**Preprocessing time:** It is the time that is needed to arrange the data for searching.
3.4.1 The complexity of computing the Voronoi diagram and Delaunay triangulation

The computation of the Delaunay triangulation of a set of $n$ points in the plane is one of the classical problems in computational geometry, and plenty of algorithms have been proposed to solve it. The most popular algorithms are given in Table 3.3.

<table>
<thead>
<tr>
<th>Name of algorithm</th>
<th>Worst case</th>
<th>Uniform data</th>
<th>Reference</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sweep Line</td>
<td>$O(n \log n)$</td>
<td></td>
<td>Fortune (1995)</td>
<td>Static</td>
</tr>
<tr>
<td>Divide-and-Conquer</td>
<td>$O(n \log n)$</td>
<td>$O(n)$</td>
<td>Guibas, and Stolfi (1985)</td>
<td>Static</td>
</tr>
<tr>
<td>Radial Sweep</td>
<td>$O(n \log n)$</td>
<td></td>
<td>Midtbø (1993)</td>
<td>Static</td>
</tr>
<tr>
<td>Randomized Incremental</td>
<td>$O(n \log n) \sim O(n^2)$</td>
<td></td>
<td>Guibas, and Stolfi (1985)</td>
<td>Dynamic</td>
</tr>
</tbody>
</table>

Table 3.3. Delaunay triangulation algorithms and the computational complexity.

An excellent presentation of a few methods is given in Midtbø (1993). In general the Divide-and-Conquer algorithm is preferred since it has a favorable running time and it is also suitable for parallel processing. However, in this work we use the Sweep Line algorithm which has the same worst case running time of $O(n \log n)$. Fortune (1995) developed a direct Delaunay triangulation implementation using the Sweep Line method. However, we will describe the less trivial algorithm of constructing a Voronoi diagram as developed by Fortune (1987) and also described in de Berg (1997, pp.145-161). The principles of both algorithms from Fortune (1995) and Fortune (1987) are the same, and we will see that the Delaunay triangulation can be computed from the resultant Voronoi diagram in linear time. The planar sweep line algorithm uses a horizontal line that sweeps from the upper point of our point set to the lower point. While the sweep is performed we draw the "beach line" or the locus of points that are closer to some site $P_i$ than to other
points. This "beach line" is a sequence of parabolic arcs (Figure 3.11), each arc being a function of the current sites. The parabolic function is given by:

\[ y = \frac{1}{2(P_{j,y} - l_y)} \cdot (x - P_{j,x})^2 + \frac{(P_{j,y} + l_y)}{2} \]  

(3.7.2)

where:

- \( P_{j,x} \) – New site's x-coordinate,
- \( P_{j,y} \) – New site's y-coordinate,
- \( l_y \) – The y-coordinate of the sweep line,
- \( x, y \) the coordinates of the parabolic arc.

![Figure 3.11: Fortune's algorithm figure taken from de Berg (1997).](image)

Thus during the sweep line process we maintain a priority queue with the following information:

**New site event**: A new arc can appear on the "beach line", through a site event.

**Circle event**: An existing arc (\( a' \) in Figure 3.12) can disappear from the "beach line" through a circle event, i.e. there is a circle passing through the sites \( P_h, P_j, \) and \( P_k \) with \( q \) (a "beach line" point which will become a Voronoi vertex) as its center and with the lowest point lying on \( l_y \) (this circle is actually tangent to the sweep line, see Figure 3.12 in the middle). The equation of a circle passing through three points is given in (3.1)
when we equate the determinant to zero. Given the information about this circle, we can write the equation of the tangent to this circle at $P_1(X_1,Y_1)$ as:

$$ (X-X_q)(X_1-X_q) + (Y- Y_q)(Y_1- Y_q) = r^2 $$  \hspace{1cm} (3.8) \\

where $r$ is the radius of the circle and $X_q$, $Y_q$ are planar coordinates of the center $q$ of the circle.

*Figure 3.12: A circle event sequence adapted from de Berg (1997). Remark in this Figure the points are marked by small letter $p_i, p_j, p_k$ unlike our previous discussion.*

Let us summarize the Sweep Line algorithm (omitting the computation of the convex hull's infinite edges):

1. Initialize the event queue $Q$ with all site events.
2. while $Q$ is not empty
3. do Consider the event with largest $y$-coordinate in $Q$.
4. if the event is a site event, occurring at site $p_i$
5. then HANDLE_SITE_EVENT$(p_i)$
6. else HANDLE_CIRCLE_EVENT$(p^*)$, where $p^*$ is the lowest point of the circle causing the event
7. Remove the event from $Q$. 

76
where P is a set of input points; procedures $HANDLE\_SITE\_EVENT$ and $HANDLE\_CIRCLE\_EVENT$ perform the insertion of a new site with its arc to the "beach line", and checking for circle event and deleting the appropriate arc, respectively.

**Complexity analysis:**

Before we prove that the Sweep Line algorithm is $O(n \cdot \log n)$ it is important to mention that the problem of computing a Voronoi diagram is reducible to the problem of sorting n real number (which has been proved to be $O(n \cdot \log n)$), so any Voronoi algorithm must take at least $\Omega(n \log n)$ (lower boundary) time in the worst case. Therefore the Sweep Line algorithm is optimal (de Berg et al. 1997).

Let us describe the components of our algorithm: Firstly the "beach line" or status structure which is represented by a balanced binary search tree, $T$. Each leaf of the data structure represents an arc, and the internal nodes of $T$ represent breakpoints on the "beach line". Secondly, the event queue $Q$ is implemented as a priority queue, and we store the upcoming events (new site events and a circle event). With those components we can analyze the algorithm: The primitive operations on our data $T$ and the event queue $Q$ operations, such as inserting or deleting an element, take $O(\log n)$ time each. The primitive operations for recording the edges in the list take constant time. To handle an event (new site or circle), we have to do a constant number of such primitive operations; so we spend $O(\log n)$ time to process an event. Since there are $n$ site events and at most $(2n-5)$ circle events we can conclude that the complexity of our algorithm is $O(n \log n)$ in time, corresponding to $O(n+2n-5)$ event *$O(\log n)$ time, and it uses $O(n)$ for storage.
To make the link between the Voronoi diagram and the Delaunay triangulation, we define the so-called Delaunay graph which has a node for every point (or Voronoi cell) and has an arc between two nodes if the corresponding cells share an edge. If \( \mathbb{R}^2 \) has no points which lie on the same circle, i.e. the determinant in equation (3.1) equals non-zero for any other point in \( \mathbb{R}^2 \) (contrary to the example in Figure 3.13), then the Delaunay graph equals the Delaunay triangulation. Otherwise, to obtain a Delaunay triangulation from the Delaunay graph, we need to triangulate the faces, which have more than three vertices. This is a trivial task since all the faces of the Delaunay graph are convex. Consequently, we can get the Delaunay triangulation from the Voronoi diagram in linear time, \( O(n) \).

![Figure 3.13: Four points that lie on the same circle, and their Voronoi diagram](image)

### 3.4.2 Complexity of points in a planar subdivision algorithm

Theoretical research has produced a number of algorithms to locate a point in a planar straight-line graph, some of which are given in Preparata and Shamos (1985, pp. 45-67) or Snoeyink (1997). Those techniques perform in \( O(n \log n) \) time to preprocess a planar
subdivision with \( n \) vertices and in \( O(\log n) \) time for queries using \( O(n) \) space. The simplest algorithm is the Slab Method which include the following steps:

1. We draw a vertical line through each of the vertices, thereby dividing the plane into \( n+1 \) slabs.
2. We sort these slabs by their \( y \)-coordinates as part of the preprocessing. This step enables us to find, using a binary tree, in \( O(\log n) \) time, the slab in which the query point \( Q \) lies.
3. We order the segments within the slab from left to right. This is used in order to find the location within the slab. Thus, we may use a binary search to determine in \( O(\log n) \) time, the trapezoid in which point \( Q \) falls.

Note that preprocessing or sorting the slabs and the segments within a slab, using a balanced binary search tree will take \( O(n \log n) \) time with \( O(n^2) \) storage space. This process can still be improved to linear time algorithm; see the details in Snoeyink (1997, p. 562). The Delaunay triangulation is a special case of a planar straight-line graph. Since we store the adjacency relationship of the Delaunay graph during the construction of it, we can use a faster method to locate the triangle in which the query point \( Q \) is located. This algorithm is called Subdivision Walking; it is described in Green and Sibson (1977) and Guibas and Stolfi (1985). The idea is to start at some arbitrary place on the subdivision and then move, one edge at a time in the general direction of the point \( Q \). In general, in a systematic point search, it is wise to take the previous point location as the starting point and, if nothing systematic is known about the position of the point, it is sensible to start from the center of our data set. One
would expect that for each point this would result in an $O(n^{1/2})$ average time and in a worst case time of $O(n)$.

![Figure 3.14: Walking method to find the triangle which includes the query point. The point in the center is the target location and we start arbitrarily with edge 12-13. We walk to 12-16, 12-17, 12-11, 11-17, 11-24, 24-17, 24-21, 21-17 until the first time that we have to repeat the same edge 24-17.](image)

**3.4.3 Complexity of the nearest neighbor search algorithm**

Smid (1999) reviewed the algorithms and data structures for proximity problems on point sets in $R^d$. For the purpose of this work it will suffice to analyze a simpler case, sometimes called "the post office problem" in the literature:
Given a set $S$ of $n$ points in $R^d$, store it in a data structure such that for any query point $P \in R^d$, we can efficiently find its nearest neighbor, i.e. a point $P^* \in S$ that is closest to $P$: 

$$d(P, P^*) = \min\{d(P, Q) : Q \in S\}$$

We will confine ourselves to a planar (2-dimensional) graph and will not discuss the whole variety of methods such as the K-D tree or the Quadtree; the interested reader is referred to Al-Daoud and Roberts (1996). Instead we will describe the Cell Method which is supposed to give an optimal constant expected-time for uniformly distributed data and was initially developed by Bentley et al. (1980). The idea of the cell method is to divide the data space into small squares, called cells or bins, of area $C/n$. $C_n$ is the density or the expected number of points in each cell ($n$ is the total number of points). This step is easily done by creating an array of size $(n/C)^2$ by $(n/C)^2$ which holds pointers to the lists of points in each bin. To find the nearest neighbor, a search is performed by starting at the cell holding the data point, and then proceeding in a spiral-like pattern to the cells surrounding it, until a point in a non-empty cell is found. Thereafter, only cells which have the same distance as the point that was found will be searched (namely, cells that intersect the circle with radius equal to the distance to the found point). Figure 3.15 demonstrates the search process with an integer denoting the order in which it was visited. This algorithm is used to find a nearest neighbor point; we will look how it can be modified to an Octant Search algorithm. Nevertheless, we will first analyze the complexity of the cell method algorithm for a uniformly distributed point data set; Naturally the preprocessing, which assigns each point to the appropriate cell can be accomplished in linear time $O(n)$ (where the assignment is a single operation performed
in constant time). The actual algorithm time is given by the following theorem from Bentley et al. (1980): “If \( n \) points are chosen independently from a uniform distribution on the unit square, then spiral search finds the nearest neighbor of a query point in constant expected time”, say \( O(1) \). The proof of this theorem as given in the paper is rather tedious, but can as well be derived using the following relationships:

- The probability of a point being placed in a certain fixed cell: \( q_p = C_n/n \)
- The probability that a particular cell is empty: \( (1-q_p)^n = (1 - C_n/n)^n < e^{-C} \)
  (using the inequality \( (1 - x/n)^n < e^{-x} \)).

Define \( K_c \) as the number of cells in each layer surrounding the query point cell then.

- Each layer contains \( 8(K_c - 1) \) cells.
- The number of cells inside and including the layer is \( (2K_c - 1)^2 \leq 4K_c^2 \).

Using the cell method we can design an algorithm for an octant search. Liu (1999, p. 33) uses an exhaustive search to find the 8 points of the octant scheme. If we use only the cell in the octant zones where no point was found, we would improve the efficiency of the algorithm.
Figure 3.15: Spiral search of nearest neighbor and quad neighbor using the cell Method.

The closest point lies in cell 9 but requires examination of cells 1-15; for the octant search we will proceed only with the cells which are in the required zones (16a-25a).

3.4.4 Comparing the complexity search methods, with a note on Hessami et al. (2001).

The problem of accurately assessing geometric algorithms arises again and again, Tipper (1995) described the problem by commenting on the linear time Delaunay triangulation by Tsai (1993). In Hessami et al. (2001) a Kriging algorithm is presented that uses Delaunay data structure for a neighborhood search method. The idea, in general, is worth pursuing and certainly possesses some advantages over other methods of neighborhood search. The algorithm as presented seems to be elegant and uses appropriate methods. It is unfortunate that the programs are not available in contrast to what was published in the paper. Moreover, the results of the experimental part may give a false impression by emphasizing the wrong reasons to use this method.
The algorithm by Hessami et al. (2001) has three components:

1. Construction of the Delaunay triangulation which is performed using a Sweep-Line algorithm in $O(n \log n)$ time.

2. Application of the Walking Method to locate the triangle which includes the predicted point; this algorithm operates in $O(n^{1/2})$ average time and in a worst case time of $O(n)$.

3. Performing the Kriging itself which is a constant time operation when we use only a local neighborhood.

Thus the algorithm performs with preprocessing time of $O(n \log n)$ and query time of $O(n^{1/2})$. Hessami et al (2001) compared their algorithm to a circular search window method and concluded that "Delaunay is considerably faster than the classical method, for an about equivalent precision". Indeed, there is no doubt that the Delaunay search method gives a better neighborhood for interpolation since the maximization-of-the-minimum-angle property creates an interpolation problem while the circle search criterion will lead to an extrapolation problem. However, it is very unexpected that their algorithm is so much faster than a simple neighborhood search technique which, as we saw in Section 3.4.3, can even be performed in preprocessing time of $O(n)$ with constant query time as proved by Bentley et al. (1980). The weak performance of the circle search algorithm may be advocated to the Kriging function which in the case of circle search gets more points as an input. The Kriging function generally operates in $O(n_i^3)$ time ($n_i$ is the number of neighborhood points), and a large number of points may influence the performance of the program.
3.5 Applying the Delaunay structure as a search method: Recursive algorithm

In this section we describe the final algorithm and programs that we used for the Delaunay-triangulation structured kriging. We developed the programs using MATLAB®'s developing environment; MATLAB is a registered trademark of The MathWorks, Inc. MATLAB has a few toolboxes which provide functions in statistics, image processing and linear algebra and, most importantly MATLAB code can be compiled to C++ and runs as a standalone application. However, MATLAB lacks a geostatistical toolbox; therefore, some basic tools had to be developed for which we used as a reference, Middleton (2000) who provides some MATLAB code. A flow chart of our interpolation process including the applied programs is given in Table 3.4. A component analysis of the computational complexity of the developed programs suggests that our preprocessing time is $O(n \cdot \log n)$ which includes building the Delaunay triangulation, and a processing time with a worst case of $O(n \cdot n_i)$ where $n_i$ is the number of locations that need to be interpolated. That includes the search for enclosing triangle function, the creation of point neighborhood and the Kriging interpolation. In the processing step, the slowest program is that for searching the enclosed triangle which takes a $O(n)$ time in the worst case. This time can be significantly improved (to $O(1)$) if the required points are not in arbitrary locations, but points with a regular pattern like a grid. Using this premise, we can almost eliminate the required search time by searching in the locality (local triangular network) of the previously interpolated point.

85
Table 3.4: flow-chart of the developed interpolation process.
CHAPTER 4
ON THE NEIGHBORHOOD OF KRIGING
INTERPOLATION

In Chapter 3 we saw how the definition of the interpolation neighborhood affects the computational burden of the algorithm. In this chapter, we examine the same problem from a statistical point of view and our concern will be the accuracy of the interpolation or the Mean Squared Prediction Error (MSPE). As we will see in the first section, neighborhood selection is a fundamental theme in statistics although, surprisingly, it did not get enough attention in the geostatistical literature, and related research is mostly empirical. We will assess the importance of the spatial configuration of the data on the resulting MSPE. This will be linked to Chapter 3 with the discussion about spatial data structure. We use Fourier analysis with some lenient assumptions to try and understand the effect of the interpolation neighborhood on our result. Lastly, we present results that we obtained from applying the developed program to the case studies data.

4.1 Introduction

4.1.1 Neighborhood and Markov Random Field

The notion of a statistical neighborhood is well established in lattice model applications, e.g. image analysis, with the significant theory of Markov Random Fields (see for example Cressie and Davidson (1998), Besag (1974), Geman and Geman (1984)). We
will briefly review the concept of neighborhoods as applied in this theory: see also Cressie (1993, pp. 410-425). Let the data point vector

\[ y = [y(s_1), ..., y(s_n)]' \]

be collected at known sites \( \{s_1, ..., s_n\} \subset S \).

We denote the joint probability distribution of the point data by \( \Pr[y(s_1), ..., y(s_n)] \)

and the conditional probabilities by \( \Pr[y(s_i)|\{y(s_j) : j \neq i\}] \) for \( i=1, ..., n \).

We specify a neighborhood \( N_i \) to be a certain set of sites \( s_j (j \neq i) \) neighboring \( s_i \) such that:

\[ \Pr[y(s_i)|\{y(s_j) : j \neq i\}] = \Pr[y(s_i)|\{y(s_j) : s_j \in N_i\}] \text{ for every } i=1, ..., n \quad (4.1) \]

Equation (4.1) implies that the probability of \( y \) at the \( i^{th} \) site \( s_i \) (given all other sites) depends only on its neighboring values \( \{y(s_j) : s_j \in N_i\} \). This neighborhood relationship should have the following properties:

1. A site is not neighboring to itself: \( s_i \notin N_i \);
2. The neighboring relationship is mutual: \( s_i \in N_j \Leftrightarrow s_j \in N_i \)

Notice that this neighborhood relationship can also be applied to irregularly spaced data where the neighbor set \( N_i \) of \( s_i \) is defined as the set of nearby sites within a fixed radius \( r \)

\[ N_i = \{s_i \in S| \text{dist}(s_i, s_j) \leq r, s_j \neq s_i\} \]

and where \( \text{dist}(s_i, s_j) \) denotes the Euclidean distance between the sites \( s_i \) and \( s_j \) (sites at or near the boundaries have fewer neighbors). A random stochastic process \( y \) is said to be a \textit{Markov Random Field (MRF)} if for any sites \( S_i \) and \( S_j \) the conditional probability on the neighborhood \( N_i \) of the \( i^{th} \) site is the same as on the group of all other sites \( s_j (j \neq i) \).

Using (4.1) and the conditional probability rules we can write:
The use of the Markov Random Field assumption means that the modeling can be carried out at the local level. A Markov Chain is said to be of order "k" if the following equation relating the conditional probabilities is satisfied:

$$\Pr \{y_{n-1}, y_{n-2}, \ldots \} = \Pr \{y_n|y_{n-1}, y_{n-2}, \ldots, y_{n-k}\}$$

for \(k\) as the smallest possible integer. The Markov Random Field theory has been also applied in spatial point processes (see for example Cressie 1993, pp. 673-681). A spatial point process is defined to be a Markov process of range \(\rho\) if the conditional probability at \(s_i\), given the realization of the process, depends only on the events in \(b(s_i, \rho)\), where \(b(s, \rho)\) is the closed ball of radius \(\rho\) centered at \(s_i\). Besag (1974) suggested to use this theory for an irregular geostatistical model which he named non-lattice system. In his classical paper he proposed the use of Voronoi diagrams (see Chapter 3) to choose the appropriate neighborhood. Li (1995, p. 5) proposed to use Delaunay triangulation to define the neighborhood of a MRF with irregular data, but he did not extend the discussion on that topic.

A couple of rules have been proposed to determine the order of the chain (4.3) or the size of the Markov neighborhood from the given observation vector \(y\). Those rules are generic statistical tools, and consequently, in this context, the neighborhood is referred to as the statistical model. They provide a measure of preference for some data models over others, and they are often an attractive alternative to hypothesis testing. Typical rules assign models a score determined by the likelihood function associated with the model,
the number of parameters (or dimension) of the model, and the data. Common rules include the Akaike Information Criterion (Akaike, 1974), the Bayes Information Criterion (Schwartz, 1978), and the Minimum Description Length (Rissanen, 1978) The principal difficulties of applying those rules to select the Kriging neighborhood are as follows: These scoring rules are based on the "given" prior probability distribution for the chosen model; this prior probability is often unknown. There are uncertainties associated with scores since two different samples of the same size from the same distribution may yield not only different numerical values for the same model, but even different orderings of models (Glymour et al. 1997), and it is computationally almost impossible on a large model space to calculate scores for all models.

Another known criterion for model selection is Mallows's $C_p$-statistic (Gilmour, 1996). This method compares different models based on the Sum of Squared Errors (SSE), the error variance, the number of observations and the number of parameters of the model being considered. Nevertheless, this method requires extensive computations and was never tested to define the Markov or Kriging neighborhood to the best of our knowledge. As a result of all those drawbacks we have looked for another criterion to choose our Kriging neighborhood. In the next section we will review research conducted in the geostatistical community regarding this topic; based on that, we will develop a new approach to look at this topic.

4.1.2 Finite domain Kriging

In general, almost all the applications of Kriging are performed in a finite domain, It is quite a surprise that this fact is hardly represented in the literature. Deutsch (1993, 1994)
reported on the consequence of Kriging of data contiguously aligned along finite strings; the weights assigned to the end points of a string are large since the end points inform about the infinite half space beyond the string. As shown in Figure 4.1, the edge samples receive a significantly greater weight than central samples. The overweighing becomes less important as the point being estimated gets closer to the string and as the relative nugget effect increases (this effect is given in detail in Deutsch, 1993). The reason for the edge samples to be overweighed is that the covariance matrix of the observations (K matrix in Section 2.3) perceive such samples as less redundant than the more centrally located samples. In those papers Deutsch (1993, 1994) described a number of empirical solutions to this problem, for example by wrapping each finite string of observation when building the covariance matrix (K) for Kriging or by using a another spatial redundancy measure in place of the covariance function to form K. This method corrects the string effect at the cost of losing the exactitude property (accuracy) of Kriging without wise. Deutsch (1993, 1994) stated that the effect is not pronounced with Simple Kriging and suggested a method that will combine the Ordinary Kriging estimate of the local mean and the Simple Kriging weights. Moreover, he also claimed that the string effect is much attenuated when many strings are involved.
Figure 4.1: Kriging weights of Ordinary Kriging as a function of the distance from the data string. The string length is one unit, the nugget effect is 20%, the range of the variogram is one unit. The proximity to the central samples becomes more important than the declustering as the point being interpolated gets closer. (taken from Deutsch, 1993)

To conclude, it seems to us that the use of empirical methods can be avoided by taking a few measures:
• Careful detrending of the data so that the local mean will be close to zero i.e. Ordinary Kriging will be close to Simple Kriging with zero mean.

• Avoiding any clusters or strings of data by choosing the points by means of Delaunay triangulation structure, which discourages the selection of clustered neighborhoods (see Section 3.2).

• Aiming at the selection of points, which will be within the variogram range; otherwise we can expect difficulties.

Deutsch (1996) described the problem of negative Kriging weights that arises often in a finite domain when data which are close to the location being interpolated, screen data from farther away. Such negative weights may occasionally lead to negative and nonphysical estimates. Those estimates are said to be nonconvex because they lie outside the range of data values used. Deutsch (1996) outlined a practical solution to this problem, namely to assign zeros to the "abnormal" weights. The Delaunay structured Kriging may lessen the amount of data screening since Delaunay triangulation favors equal representation in different directions. Krige (1996, 1997) opposed the methods, taken by many practitioners that aim to produce a set of Kriged block estimates with little or no smoothing by manipulating the data search routine. He contended that such estimates are conditionally biased, particularly when the sample size \( n \) is small. However, a few experiments presented in his first paper suggested that there is a value for the number of points (or a search distance ) beyond which, by adding more points, the Kriging result will not be changed. Vecchia (1988, 1992) developed a maximum
likelihood estimation method based on the assumption that the underlying process is Gaussian. In applying his method he used observations, which are closest to the interpolated point in the sense of the ordinary Euclidean distance. His experiments demonstrated that, for a dense point distribution, the estimation results remain stable for the number of neighborhood points \( N_i \geq 10 \), and in case of a less dense dataset for \( N_i \geq 6 \).

A few papers addressed the problem of the discontinuity which is a result of Kriging in a local neighborhood, and a few ad-hoc methods were proposed. For example, Aunon and Gomez-Hernandez (2000) suggested a method to patch together dual Kriging estimates, obtained with data from different neighborhoods, by means of a blending belt around each neighborhood. They used an interpolating polynomial in the belt with boundary conditions to ensure continuity. Chilès and Delfiner (1999, pp. 201-209) suggested a few techniques to improve continuity without spoiling the accuracy of the estimate. A simple technique is to use a selective low-pass filter which modifies the estimates within the Kriging confidence interval as defined by the Kriging standard deviation. Chilès and Delfiner (1999, pp. 205-206) also defied the common assertion that, when a variogram has a finite range, it is not necessary to include sample points beyond that range in the Kriging neighborhood. This fact is understandable for Ordinary Kriging since it includes an estimation for the mean which is not a local phenomenon. However, they claimed it also for Simple Kriging since points beyond the range distance may still exert influence through their correlation with points within the range by virtue of the so-called “relay effect”. Probably the most informative discussion on the topic is given in Cressie (1993, pp. 131-134) where it is argued that the range by itself does not provide sufficient
guidance to determine the Kriging neighborhood. This fact is demonstrated by a simple numerical example. Moreover, it is shown that, often, Kriging neighborhoods have to be expanded as the nugget effect increases (relative to the sill) since the screen effect gets weaker. On the same issue, Cressie (1993, pp. 176-177) suggested to view Kriging as a two-stage problem, i.e. the mean/trend estimation stage and thereafter the choice of the neighborhood depend only on the $n_k$ Simple Kriging weights in the vector $L = \kappa(s')^T n_k K^{-1} n_k$, where $n_k$ denotes the number of data locations in the neighborhood. Cressie (1993, pp. 158-160) reported on a practical way to choose the neighborhood of Kriging by increasing gradually the neighborhood while computing each time the optimal weights (see Section 2.3). If it exists, the neighborhood with the fewest points for which $x(s_0) \approx \kappa(s_0)^T K^{-1} y$ is chosen. In the given example, the nugget effect was large and therefore most of the data locations were required; thus a compromise had to be taken. It was noted that "this compromise between kriging variance and size of Kriging neighborhood is worthy of more research". To summarize this literature review, we will quote Olea (1999, p. 63) who said that ". the neighborhood must be large enough to contain three observations at a bare minimum, and anything beyond 25 observations is considered more than adequate... More precise justifications for selection of neighborhood size depend upon the measure of performance achieved and fluctuate according to the nature of the sampling pattern and the covariance. Lacking theoretical criteria, experiments ... are the best alternative." In the next section we will examine some parameters for neighborhood selection and in the subsequent section we will attempt to scrutinize our data using Fourier analysis.
4.2 Parameters for neighborhood selection

In this section we will look at the different parameters that should influence neighborhood selection. Clearly among those factor are the following:

- Avoiding extrapolation.
- Selecting the closest points as our neighborhood.
- Maintaining continuity (overlapping zone).

We will not discuss the two last factors which are rather trivial in view of the definition of Kriging and the MRF theory. However, we will demonstrate the critical effect of extrapolation or point configuration by a simple example. Bucher and Vckovski (1995) gave the following definition:

**Interpolation:** Interpolation aims at predicting values "in between" sampled values, i.e. to increase the resolution of an existing information source. The process value at $s'$ is found by interpolation, if $s'$ is within the convex hull of all sites $s_i$ while $s'$ and all $s_i$ are disjunctive, i.e. $s' \in \{ s_i | i=1,\ldots,n \}$.

**Extrapolation:** It is similar to interpolation except that the location of $s'$ is not within the convex hull of the sites $s_1,\ldots,s_n$.

We will look at a 1-dimensional example, and we will find the MSPE of the interpolated value as a function of its location using both Ordinary Kriging and least-squares adjustment fit of a first order polynomial function.
Figure 4.2: Example to compare interpolation and extrapolation. $Y_3$ at $X_3$ is interpolated while $Y'_3$ at $X'_3$ is extrapolated.

**Example:** We are given two planar coordinates $X_1, X_2$ and two measurements of there attribute value (for example height at $X$ location) $Y_1$ and $Y_2$. We took for this example: $X_1=4; X_2=8$;

**The extrapolation error using Ordinary Kriging:** We take, for example, a simple semi-variogram model such as the linear model, given in Cressie (1993, p. 61, (2.3.7)), with "nugget", $\sigma_e^2=2.5$;
Taking the Mean Square Prediction Error formula from Cressie (1993, p. 122, (3.2.16), we obtain

\[ MSPE = \gamma_0^T \cdot \Gamma_0^{-1} \gamma_0 \]  

(4.5.1)

We construct our vector \( \gamma_0 \) and matrix \( \Gamma_0 \) using the given numbers:

\[
\Gamma_0 = \begin{bmatrix} 2.5 & 10.5 & 1 \\ 10.5 & 2.5 & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad \gamma_0 = \begin{bmatrix} 2 \cdot \|x3 - x1\| + 2.5 \\ 2 \cdot \|x3 - x1\| + 2.5 \\ 1 \end{bmatrix}
\]

(4.5.2)
The graph of the MSPE as a function of the location of $X_3$ is given in Figure 4.4 for $X_3=2$ the MSPE is 3.9375, for $X_3=6$ the MSPE is 1.5.

If we take a semi-variogram which increases slower, for example:

$$\gamma(h) = \begin{cases} 
0 & h = 0 \\
2.5 + 0.75 \cdot \|h\| & h \neq 0 
\end{cases}$$

(4.5.3)

then the ratio will be smaller and the computed results are:

For $X_3 = 2$ the MSPE is 5.5; for $X_3=6$ (point of inflection) the MSPE is 4. Naturally, the error will be bigger as we make the slope smaller. A similar graph with a power model is given in Chilès and Delfiner (1999, pp. 173-174).

Developing an understanding on the significant influence of the spatial configuration on the MSPE is vital in data analysis. Consequently, we applied the same methodology to estimate the effect of extrapolation Vs. interpolation on our magnetic data (see Section 1.3.1) with the semi-variogram as given in (2.2.30). The chosen locations in [m] are $X_1=3000$ and $X_2=5000$; and we compute the MSPE of Ordinary Kriging as a function of the location $X_3$ with the results given in Figure 4.5.
Figure 4.5: The computed MSPE as a function of $X_3$ for the nested Gaussian model of the magnetic data as given by (2.2.30). For $X_3 = 4000\text{m}$ the MSPE is $2173\ \text{nT}^2$; for $X_3 = 2000\text{m}$ the MSPE is $5647\ \text{nT}^2$.

**The extrapolation error using least-squares adjustment for polynomial fitting**

Both $Y_1$ and $Y_2$ have measurement error associated with them, $e_1$ and $e_2$, say. $X_1$ and $X_2$ are error-free. So our mathematical model is:

\[
\begin{bmatrix}
Y_1 \\
Y_2
\end{bmatrix} = \begin{bmatrix} X_1 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}, \quad \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \sim \begin{bmatrix} 0, \sigma^2 \end{bmatrix}
\]

Using least-squares adjustment and the Schur complement formula to expand the inverse matrix, we compute the dispersion matrix of the results (equal to the MSE matrix due to unbiasedness):
\[
D \left[ \begin{array}{c}
\hat{a} \\
\hat{b}
\end{array} \right] = \sigma_0^2 \cdot \begin{bmatrix}
\frac{2}{X_1^2 + X_2^2 - 2X_1X_2} & \frac{X_1 + X_2}{X_1^2 + X_2^2 - 2X_1X_2} \\
\frac{X_1 + X_2}{X_1^2 + X_2^2 - 2X_1X_2} & \frac{2}{X_1^2 + X_2^2 - 2X_1X_2}
\end{bmatrix}
\]

Using the law of error propagation and the Jacobian.

\[
D[E(Y_3)] = \sigma_0^2 \cdot \frac{2 \cdot X_3^2 - 2X_3 \cdot (X_1 - X_2) + (X_1^2 - X_2^2)}{(X_1 - X_2)^2}
\]

If we minimize the dispersion of this estimate, we arrive at:

\[
\frac{\partial D[E(Y_3)]}{\partial X_3} = 0 \Rightarrow X_3 = \frac{X_1 + X_2}{2} \quad \text{with}
\]

\[
D[E(Y_3)]_{\text{min}} = \frac{\sigma_0^2}{2} \cdot \frac{2 \cdot (X_1^2 + X_2^2) - (X_1 + X_2)^2}{(X_1 - X_2)^2} = \frac{\sigma_0^2}{2}
\]

In Figure 4.6 we took for example \( X_1 = 4, X_2 = 8, \sigma_0 = 1 \) and we plotted the error function for different values of \( X_3 \). When \( X_3 = 2 \) the MSE is 40 when \( X_3 = 6 \) (point of inflection) the MSE is 8. Due to the unbiasedness of the least-squares adjustment, the dispersion is equal to the Mean Square Error (MSE).
Figure 4.6: The effect of the extrapolation error in 1st order polynomial fitting.

\[ \text{MSE}(\hat{Y}_3) = \text{MSE}\{\hat{a} \cdot X_3 + \hat{b}_3\} = E\{((\hat{a} - a) \cdot X_3 + \hat{b}_3 - b_3)^2\} = \]
\[ = D\{(\hat{a} - a) \cdot X_3 + (\hat{b}_3 - b_3)\} + E\{((\hat{a} - a) \cdot X_3 + (\hat{b}_3 - b_3))^2\} = \]
\[ = D(\text{E}\{\hat{Y}_3\}) + 0 \]

(4.5.7)

The above examples demonstrate the well-known fact that extrapolation can result in a high prediction/estimation error; for simplicity, they are given in one dimensional space but the extension to \( n \) dimensions follows along the same lines. Extrapolation is more critical in least-squares adjustment (LSE) than in Ordinary Kriging (OK). However, even in the Kriging approach there is a great impact from the point configuration, especially on rough surfaces where the semi-variogram increases fast. Both the OK example and LESS polynomial fitting are special cases; a general proof of the extrapolation effect can be derived by using the generic form of a stationary semi-variogram (stationarity implies that the variogram reaches a sill) such as:
\[
\gamma(h) = \begin{cases} 
\sigma^2 + C_1 f(h) & 0 < h \leq r \\
\sigma^2 + C_1 f(r) & h > r \\
0 & \text{if } h = 0
\end{cases}
\] (4.5.8)

as well as a generic polynomial/function fitting in the LESS example. The OK extrapolation effect on the MSPE may well be explained by decomposing it into a local mean term (LSE) and a stochastic term; see Cressie (1993, p. 174, equation (3.4.64)) for more details.

4.3 Data analysis using Fourier theory

Fourier theory is a popular tool to examine and analyze numerical data. As part of the analysis the data are transformed from the spatial/time representation /domain to the frequency domain. Many operations, which are impossible or hard to perform on the spatial/time domain, can be carried out efficiently in the frequency domain. In Section 4.3.1 we will briefly review the Fourier analysis sampling theory - the heart of today's digital communication -, which allows us to determine the minimum required sampling interval. Subsequently, in Section 4.3.2 we will use this theory to analyze our data and determine which samples are redundant. Finally, in Section 4.3.3 we will use the frequency representation of the signal to develop some understanding about the required interpolation neighborhood. In that Section we make the link to the spatial coherency function and evaluate the relationship of the covariance to the interpolation neighborhood.
4.3.1 Detrending the data, minimum point set required

How can we know whether we have a large enough sample to represent our process reliably? Communication and information theory provided an answer to this question on a regular grid. In the first half of the 20th century, Harry Nyquist and Claude Shannon (see for example Shannon (1948)) developed a mathematical framework to determine how often a signal should be sampled. Their result, known as the sampling theorem, states that: "In order to be perfectly represented by its samples, a signal must be sampled at a sampling rate equal to at least twice its highest frequency component."

A band-limited signal is a signal, y(s), which has no spectral components beyond a frequency of B Hz; using the spectral representation, this means:

\[
y(f) = 0 \text{ for } |f| > B. \tag{4.6.1}
\]

The Sampling Theorem states that a real signal, y(s), which is band-limited to B Hz can be reconstructed without error from samples taken uniformly at a rate R>2B sample per second. This minimum sampling frequency

\[
F_s = 2B \text{ Hz} \tag{4.6.2}
\]

is called the Nyquist rate or the Nyquist frequency. The corresponding sampling interval,

\[
T = \frac{1}{2B} \tag{4.6.3}
\]

104
is called the *Nyquist interval*. A signal band-limited to \( B \) Hz which is sampled at less than
the Nyquist frequency of \( 2B \), i.e., which was sampled at an interval \( T > \left( \frac{1}{2B} \right) \), is said to
be *undersampled*.

**Aliasing** ; The sampling theorem assumes that a signal is band-limited. In practice, however, signals are space/time-limited, not band-limited. As a result, determining an adequate sampling frequency, which does not lose desired information, can be difficult. When a signal is undersampled its spectrum has overlapping tails; that is, \( y(f) \) does no longer have complete information about the spectrum, and it is no longer possible to recover \( y(s) \) from the sampled signal. In this case, the tailing spectrum does not go to zero, but is folded back onto the apparent spectrum. This inversion of the tail is called spectral folding or aliasing.

To summarize this brief discussion (for a compressive discussion of this topic see Schenk (1999, pp. 17-38)), we can state that determining the minimum sampling interval of any surface involves the following steps:

1. Select a representative profile of the surface.

2. Capture the representative profile with a very high point density.

3. Use Fourier analysis to transform the signal to the frequency domain.

4. From the spectrum (frequency representation) of the signal, evaluate the frequency beyond which there are no spectral components (\( B \) in equation (4.6.1)).
5. Compute the Nyquist frequency and the appropriate sampling interval.

4.3.2 Example: Applying the sampling theory to the magnetic data

In this example, we compute the Nyquist frequency of a profile taken from the magnetic data. Our data were sampled very densely on the vertical directions with an almost regular sampling interval of 50m between subsequent sample points.

We use, as our representative profile, flight line number 11730; this line includes 2054 points, with space of 50 meters from each other on a fairly regular pattern.

We transformed our data to the frequency domain (step 3 in the previous section):

\[ y_k(\hat{f}) = F^{-1}(y(s)) \]  

(4.6.4)

We used a Fast Fourier Transform (FFT) program for the transformation see also Figure 4.7. In order to use this algorithm we reduced the number of points to \(2^{11}=2048\), i.e. we took only 2048 sequential samples instead of the 2054 points of the original profile.

When the number of samples is a power of two, the FFT algorithm works faster. Technically speaking, in this case, the FFT algorithm works in \(O(n \cdot \log n)\) time (Middleton 2000, p. 112). Consequently, we may define:

\[ N=2048 \] (number of points taken from our profile);

\[ \Delta x = 50m \] sampling interval between subsequent points.

which means that our dense profile has a sampling frequency of:

\[ f_s = 1000m/50m = 20 \] samples per kilometer.

The power spectral density function, a measurement of the energy at various frequencies, is given by:

\[ G(f) = y_f(f) \cdot \tilde{y}_f(f) / N \]  

(4.7.1)
where $\bar{y}_f(f)$ is the complex conjugate of $y_f(f)$. This represents the frequency content of $y(f)$ in the range from zero frequency (nicknamed Direct Current -DC) up to and including the Nyquist frequency. The power spectrum graph of our magnetic data is given in Figure 4.8.

Looking at Figure 4.8, we can see that our signal is roughly band-limited with a bounding frequency of one sample per kilometer (notice that the y-axis is logarithmically scaled) and we can compute our approximate Nyquist frequency as:

$$f_N = 2f = 2 \text{ samples per kilometer.}$$  \hspace{1cm} (4.7.2)

Summing up the Fourier coefficients of our entire spectrum (the area below the graph or the signal energy) we get:

$$P_{\text{all}} = 9.91070534843754e*10^6$$

Summing up all the Fourier coefficients until our approximated Nyquist frequency we get:

$$P_N = 9.870928214053497e*10^6$$

which means that 99.59% ($P_N / P_{\text{all}}$) of our signal energy is preserved when we sample our signal at a frequency of two sample per kilometer. If we transform the Nyquist frequency of (4.7.2) to a sampling interval using (4.6.3) we get:

$$\Delta x_N = 1/2 = 0.5 \text{ km} = 500\text{m}$$
Figure 4.7: Our magnetic signal (after removing the mean) and its Fourier transform.
Figure 4.8: Power spectrum of the magnetic data (flight line 11730) with y-axis logarithmically scaled.

4.3.3 Neighborhood selection in the Fourier domain

In this section we will view the problem of neighborhood selection from a different angle; namely, we view it using Fourier theory. Where interpolating by using a local neighborhood, we effectively multiply our data by a Rect function (rectangular box) in the spatial domain. Mathematically speaking a Rect function, also called boxcar function, is defined in Schenk (1999, p. 29) by:

\[
z(s) = \begin{cases} 
1 & \text{for } -T < t < T, \\
0 & \text{Otherwise.} 
\end{cases}
\] (4.8)
Multiplication in the spatial domain is equivalent to convolution in the frequency domain with the Fourier dual of the function. The Fourier dual of the Rect function is the Sinc function defined by:

\[
Z(f) = 2T \frac{\sin(fT)}{fT}
\]  

(4.9)

The Sinc function (see Figure 4.9) oscillates at a constant rate of \(\pi/T\) and the amplitude decays by \(1/T\). The main lobe extends from \(-\pi/T\) to \(\pi/T\) and thus the wider the Rect function the narrower the central lobe and the oscillations. Looking back to our neighborhood selection problem; if we choose a bigger neighborhood we convolve our data with a narrower Sinc function. The narrower the Sinc function the less distortion we introduce to our signal. This of course agrees with the basic Kriging variance equation that implies that the larger the neighborhood the more precise our estimate.
Figure 4.9: Rect function and its frequency representation. Top a narrow Rect function (pulse) is a wide Sinc in the frequency domain (line). Bottom a wider Rect function will have a narrower Sinc function.

4.4 Experiments performed on the TAMARA case study dataset

This set of experiments has a couple of goals: Firstly we had to check the programs that were developed and described in Section 3.5. Secondly we wanted to evaluate the accuracy of the proposed Delaunay structured Kriging process. This process should also validate the spatial coherency functions that we have derived for the magnetic data set in Section 2.2.5. The developed programs interpolated the points of the test data set using the given data, and the following parameters are computed to analyze the results:
• Actual Mean Squared Prediction Error (MSPE) - computed from the test data (independent data set) by the following formula:

\[ MSPE = \frac{\sum_{i=1}^{n} (\tilde{x}_i - x_i)^2}{n}. \]

• Actual \(|\text{difference}| - The absolute difference between the true value and the predicted one.

• Mean of \(\sigma_0^2 - The mean of the interpolation variance; either the Kriging variance as computed by formula (2.3.9), or the linear interpolation formula as computed from formula (3.5.5).

• Max(MSPE) - The highest interpolation variance value.

• Max \((\tilde{x} - x)^2 - The highest squared value of the difference between the true value and the predicted value.

Experimental results: Our first data set includes 1195 aero-magnetic data points with 500m spacing in the horizontal direction and 2500m in the vertical direction. Our Test data include two vertical lines of 90 points in the middle of the test site. See Figure 4.10 for experiment configuration. Table 4.1 present the results of the experiments with the data before detrending process; Table 4.2 describe the results of the experiments with datrended data i.e. the residuals of the samples minus a 3rd order polynomial surface.
<table>
<thead>
<tr>
<th>Method</th>
<th>Actual MSPE</th>
<th>Mean MSPE/MSE</th>
<th>Max (MSPE)</th>
<th>Max ((\bar{x} - x)^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forth order Kriging</td>
<td>1528.73</td>
<td>1998.71</td>
<td>8445.87</td>
<td>20903.66</td>
</tr>
<tr>
<td>Third order Kriging</td>
<td>1133.37</td>
<td>2041.72</td>
<td>8532.20</td>
<td>11918.74</td>
</tr>
<tr>
<td>Second order Kriging</td>
<td>1291.81</td>
<td>2083.73</td>
<td>8676.49</td>
<td>16236.62</td>
</tr>
<tr>
<td>First order Kriging</td>
<td>1343.31</td>
<td>2148.41</td>
<td>9272.11</td>
<td>15506.47</td>
</tr>
<tr>
<td>Zero order Kriging</td>
<td>1342.54</td>
<td>2223.64</td>
<td>9626.86</td>
<td>17592.57</td>
</tr>
<tr>
<td>Linear interpolation</td>
<td>1446.89</td>
<td>403.36</td>
<td>718.82</td>
<td>21107.73</td>
</tr>
</tbody>
</table>

Table 4.1: Results of experiments on magnetic data with no detrending (Magdata.txt)

<table>
<thead>
<tr>
<th>Method</th>
<th>Actual MSPE</th>
<th>Mean MSPE/MSE</th>
<th>Max(MSPE)</th>
<th>Max((\bar{x} - x)^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forth order Kriging</td>
<td>1075.68</td>
<td>1223.08</td>
<td>4171.39</td>
<td>11777.45</td>
</tr>
<tr>
<td>Third order Kriging</td>
<td>1074.64</td>
<td>1231.13</td>
<td>4199.86</td>
<td>11474.90</td>
</tr>
<tr>
<td>Second order Kriging</td>
<td>1216.25</td>
<td>1242.58</td>
<td>4268.9</td>
<td>15784.90</td>
</tr>
<tr>
<td>First order Kriging</td>
<td>1301.94</td>
<td>1272.09</td>
<td>4513.4</td>
<td>17211.86</td>
</tr>
<tr>
<td>Zero order Kriging</td>
<td>1330.83</td>
<td>1310.52</td>
<td>4713.2</td>
<td>18636.61</td>
</tr>
<tr>
<td>Linear interpolation</td>
<td>1425.50</td>
<td>403.36</td>
<td>718.82</td>
<td>20506.52</td>
</tr>
</tbody>
</table>

Table 4.2: Results of experiments on magnetic data with detrending (Magdatat.txt)
Analysis of the experiments: The results that we obtained seem to fit the theory developed in Section 4.1.2. It did not surprise us that, the more we increased our interpolation neighborhood the better our predictions become. This was true until we had reached fourth-order level. At this level the stationarity assumption does not hold anymore and the results become worse than at the previous interpolation level and even worse than TIN based linear interpolation. The data, which were not detrended had a much bigger error in the fourth level Kriging since they are less stationary than the detrended data. The estimated MSPE was higher than the actual MSPE in the Table 4.1. This may be due to the high estimated nugget effect in the model.

It seems that the Delaunay structured Kriging interpolation outperforms TIN based linear interpolation in most cases and this without any additional computational burden. However, the advantage of Delaunay structured Kriging interpolation is that we can carry
out the two methods, both the Kriging and the linear interpolation using the same data structure. This ability is very important since it gives a very good indicator to the user about the quality of the interpolation. In other words, today an advanced GIS is capable of obtaining the correct Kriging interpolation parameters (sill, nugget, range etc.) almost without human intervention (see Gribov et al. 2001). The next generation of GIS should be able to also choose between the most accurate and suitable interpolation method. The software should be able to perform cross-validation with a few selected points and to compare with TIN based linear interpolation or any other method and using this comparison suggest the finest method. We propose to use TIN linear interpolation as the "gold standard" to compare any other interpolation with since it is simple to compute and the results are unique (except for the rare case of four circum-circular points). Consequently, in the case of Kriging, for example; the GIS interpolation program computes the prediction variance and then performs interpolation assessment based on a small subset of points, which will be computed both with Kriging and with TIN. The interpolation program will then report on the relative accuracy of the Kriging.
CHAPTER 5

EXPERIMENTS WITH OPTIMAL BIASED KRIGING

In this chapter we present some implementation issues and experiments with Optimal Biased Kriging. This type of Kriging gives up the unbiasedness condition to gain a better Mean Square Prediction Error (MSPE). We will review previous research aimed at reducing the MSPE using shrinkage predictors and derive the equations of an optimal linear biased Kriging predictor. Then we will investigate a major impediment in Kriging computation, namely, the inversion of the spatial coherency matrix. We will propose a new method which exploits spatial sorting techniques to create sparse matrices for efficient matrix inversion. Last, we will present results from experiments performed using the reviewed theory and the relatively new spatial coherency measure, called homeogram, also known as the non-centered covariance function.

5.1 Concept and formulas

In many earth science applications the user is interested in a map with higher relative accuracy; this is in contrast with applications where the data should be tied to a national datum or a mean base value. We take for example the TAMARA project (Csathó et al. 1999), an international effort to map the geological structure of the West Antarctic rift system from a newly acquired aero-magnetic data set. Here the scientists investigate local maps and look at relative alterations of magnetic values which indicate magnetic...
anomalies. The absolute magnetic values at those local areas are of minor importance. A few papers discuss methods, which give up the unbiasedness to obtain an improved Mean Square Prediction Error. In this work, however, we will closely follow the approach developed in Schaffrin (1993,1997,2000c and 2001); beforehand, we will briefly review other work in this subject. Gotway and Cressie (1993) constructed a large class of predictors with uniformly smaller risk (where the risk function is defined as the mean of the total sum of squared errors - MSPE) than the classical homBLUP (Best homogeneously Linear Unbiased Prediction as explained in Section 2.3. The basis of that development was the Best inhomogeneously Linear Predictor (inhomBLIP or Simple Kriging) as derived in section 2.3:

\[ \bar{x}_g(s') = \kappa(s')^T \cdot K^{-1} y + (a(s')^T - \kappa(s')^T \cdot K^{-1} A) \cdot \beta_0 \]  

(5.1.1)

where the mean functions vector was written as

\[ \begin{bmatrix} \mu(s_1) & \mu(s_2) & \cdots & \mu(s_n) \end{bmatrix} = A \cdot \beta_0 \]

while \( \mu(s') = a(s')^T \cdot \beta_0 \)

The general class of predictors by Gotway and Cressie (1993) was generated by different estimates of \( \beta \) which need to replace \( \beta_0 \) if this is unknown:

\[ \bar{x}_g(s') = \kappa(s')^T \cdot K^{-1} y + (a(s')^T - \kappa(s')^T \cdot K^{-1} A) \cdot \hat{\beta} . \]  

(5.1.2)

For example, a Universal Kriging predictor was obtained by taking \( \hat{\beta} \) to be the weighted least-squares estimator of \( \beta \). In their paper, however, Gotway and Cressie (1993) focused on predictors obtained using shrinkage estimators of \( \beta \), such as the James-Stein estimator where they took \( \hat{\beta} \) in (5.1.2) to be:
\[ \hat{\beta}_{js} = \frac{1 - b(y - A\hat{\beta}_{gs})^T \cdot K^{-1} \cdot (y - A\hat{\beta}_{gs})}{\hat{\beta}_{gs}^T \cdot A^T \cdot K^{-1} \cdot A \cdot \hat{\beta}_{gs}} \]  

(5.1.3)

\[ \hat{\beta}_{gs} = (A^T \cdot K^{-1} \cdot A)^{-1} \cdot A^T \cdot K^{-1} \cdot y \]  

(5.1.4)

where \( b \) is a positive constant that controls the amount of shrinkage of \( \hat{\beta}_{js} \) (Cressie 1993, p. 175). Schaffrin (1993) investigated homogeneous-isotropic processes on the sphere. Those processes cannot be both Gaussian and ergodic. A non-ergodic process does not allow us to equivalently express "expectation" and "covariance" as spatial integral over any of its realizations. Consequently Schaffrin (1993), independently, proposed and presented the formulas for the Optimal Biased Kriging predictor (also termed homBLIP - Best homogeneously LLinear Prediction). The homBLIP is a biased alternative to Ordinary Kriging with a slightly reduced mean square error. Furthermore, Schaffrin (1993) developed a complete set of kriging equations, including the MSPE computation for this case and used a new and superior type of spatial coherency function, namely, the homeogram. We have already presented some arguments that support the use of the homeogram as a better choice of a spatial coherency function. Here we will see that it also simplifies the Optimal Biased Kriging formulas and the prediction algorithm. Using the same model as before, (2.3.1), we derive the homBLIP predictor as follows:

Given the standard assumption of no correlation between the measurement error \( e \) and the process random variation \( e_0 \) which entail no correlation with the process \( x \) itself, namely:

\[ C(e,e_0) = 0 \Rightarrow C(x,e) = 0, \]

we apply the properties of the homBLIP of \( x' = x(s') \) as follows:
(1) Homogeneously linear: \( \tilde{x}(s') = L_1y \) 

where \( L_1 \) is the required linear function.

(2) Best or minimum trace of the MSPE matrix (i.e. minimum sum of mean squared errors):

\[
\text{trMSPE}(\tilde{x}) = \text{tr}D(\tilde{x} - x') + E\{\tilde{x} - x'\}^T \cdot E\{\tilde{x} - x'\} = \text{tr}(L_1y - x')^T \cdot E\{L_1y - x'\}
\]

\[
= \text{tr}\{D(L_1(x + \epsilon) - x') + \beta(L_1\tau - L_1\tau - 1)^T \cdot (L_1\tau - L_1\tau - 1)\beta\}
\]

\[
= \text{tr}\{D[L_1(x + \epsilon) + L_1(x - x')] + (L_1\tau - L_1\tau - 1)^T\beta^2 (L_1\tau - L_1\tau - 1)^T\}
\]

\[
= \text{tr}\{L_1 \cdot C(x, e) \cdot L_1^T + L_1 \cdot C(x, x) \cdot L_1^T - L_1 \cdot C(x, x') - C(x', x) \cdot L_1^T + C(x', x') + (L_1\tau - L_1\tau - 1)^T\beta^2 (L_1\tau - L_1\tau - 1)^T\}
\]

There is no correlation between the process and the measurement error:

\[ C(x', e) = 0; \quad C(x, e) = 0. \]

We will now use spatial notation and, for brevity, we will write:

\[
\kappa(s') := [C(x(s_1), x(s'))) \ldots C(x(s_n), x(s'))) \]^T. \quad (5.1.7)
\]

If we assume that the covariance function is isotropic and stationary, then:

\[
\kappa(x, x') = C(x', x) = [C(x(s_1), x(s'))) \ldots C(x(s_n), x(s'))) \]^T = \kappa(s'), \quad i \in \{1, \ldots, n\};
\]

\[
C(x', x) = C(x(s'), x(s')) = C(0) = \sigma^2_x = C_x(s', s');
\]

and from (2.3.1):

\[
C(e, e) = I\sigma^2_e; \quad C(x, x) = C_x.
\]

Therefore:

\[
\text{tr}(\text{MSPE}(\tilde{x})) = \text{tr}\{L_1 \cdot I\sigma^2_e \cdot L_1^T + L_1 \cdot C_x \cdot L_1^T - \kappa(s') \cdot L_1^T - L_1 \cdot \kappa(s') + C_x(s', s') + (L_1\tau - L_1\tau - 1)^T\beta^2 (L_1\tau - L_1\tau - 1)^T\}
\]

which we consider as our target function \( \Phi(L_1^T) \) that need to be minimized:

\[(5.1.6.2)\]
\[ \Phi(L_i^T) = \text{tr}\{L_i^T (L_i^2 + C_x) \cdot L_i^T - \kappa(s')^T \cdot L_i^T - L_i \cdot \kappa(s') + \sigma \cdot \tau \cdot (L_i^2 + \tau) \} \]

The Euler – Lagrange necessary conditions (first partial derivatives set to zero) are:

\[ \frac{\partial \Phi(L_i^T)}{2 \cdot \partial L_i^T} = [L_i^2 + (C_x + \tau \cdot \beta^2 \cdot \tau^T)] \cdot L_i^T - (\kappa(s') + \tau \beta^2) = 0 \quad (5.1.6.3) \]

and, after rearranging equation (5.1.8), we get:

\[ L_i^T = (L_i^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T)^{-1} (\kappa(s') + \tau \beta^2) \quad (5.1.7) \]

\[ \bar{\eta}(s') = L_i y = (\kappa(s') + \tau \beta^2)^T \cdot (L_i^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T)^{-1} y \quad (5.1.8) \]

Given the definition of the homeogram function in Section 2.2 we introduce:

\[ \eta(s') = \kappa(s') + \tau \beta^2 \quad (5.1.9) \]

\[ H = (L_i^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T) \]

with

\[ \eta(s') = [\eta(s', s_1), \ldots, \eta(s', s_n)]^T, \quad (5.1.10) \]

\[ H = [\eta(s_y)] + \sigma \cdot I_n; \quad i, j \in \{1, \ldots, n\}. \]

We directly obtain:

\[ \bar{\eta}(s') = \eta(s')^T \cdot H^{-1} y \quad (5.1.11) \]

since

\[ L_i = \eta(s')^T \cdot H^{-1}. \]

It is easy to check that the sufficient condition is satisfied and the second derivative is positive-definite:

\[ \frac{\partial^2 \Phi(L_i^T)}{2 \cdot \partial L_i \cdot \partial L_i^T} = L_i^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T > 0. \quad (5.1.12) \]
The MSPE of this prediction is a number and, according to (5.1.6), we compute:

\[
\text{MSPE}(\tilde{x}) = L_1 \cdot C(e, e) \cdot L_1^T + L_1 \cdot C(x, x) \cdot L_1^T - L_1 \cdot C(x, x') - C_s(x', x) \cdot L_1^T + C(x', x') + \{L_1 \tau - 1\}^2 \cdot \beta^2
\]

\[
= L_1 \cdot (I \sigma^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T) \cdot L_1^T - L_1 \cdot \{\kappa(s') + \tau \cdot \beta^2\}...
\]

\[
\cdot L_1^T \cdot \sigma^2 + \beta^2 =
\]

\[
= \eta(s')^T \cdot H^{-1} \cdot (I \sigma^2 + C_x + \tau \cdot \beta^2 \cdot \tau^T) \cdot H^{-1} \eta(s') + \{\sigma^2 + \beta^2\} -...
\]

\[
- \eta(s')^T \cdot H^{-1} \{\kappa(s') + \tau \beta^2\} - \{\kappa(s')^T + \beta^2 \tau^T\} H^{-1} \eta(s')
\]

\[
= \eta(s')^T \cdot H^{-1} \eta(s') + \eta(0) - \eta(s')^T \cdot H^{-1} \eta(s') - \eta(s') H^{-1} \eta(s')^T = \eta(0) - \eta(s')^T H^{-1} \eta(s')
\]

and finally:

\[
\text{MSPE}(\tilde{x}(s')) = \eta(0) - \eta(s')^T H^{-1} \eta(s').
\]  \hspace{1cm} (5.1.13)

There are two ways to compute the solution: firstly, by using the primal system as defined by:

\[
\tilde{x}(s')^T = \eta(s')^T \cdot H^{-1}
\]

\[
\tilde{x}(s') = \tilde{x}(s')^T \cdot y
\]

and

\[
\text{MSPE}(\tilde{x}(s')) = \eta(0) - \tilde{x}(s')^T \cdot \eta(s');
\]  \hspace{1cm} (5.1.14)

alternatively the dual system is given by:

\[
\bar{g} = H^{-1} \cdot y
\]

\[
\tilde{x}(s') = \eta(s')^T \cdot \bar{g}
\]

with no direct access to the mean square prediction error.

Schaffrin (2001) developed equivalent systems for the Optimal Biased Kriging with the covariance function and with the semi-variogram function. Comparing the various sets of
equations brings about the advantage in terms of MSPE. In Appendix B we have derived the homBLIP equations in terms of the covariance. The principle formulas are given below to demonstrate the improvement in the MSPE.

\[
\tilde{x}(s') = \hat{\beta} + \kappa (s')^T K^{-1} (y - \tau \cdot \hat{\beta})
\]

with

\[
\hat{\beta} = (\beta^{-2} + \tau^T K^{-1} \tau)^{-1} (\tau^T K^{-1} y)
\]

and the MSPE is

\[
\text{MSPE}(\tilde{x}(s')) = \sigma_y^2 - \kappa (s')^T K^{-1} \kappa (s') + \frac{(1 - \kappa (s')^T K^{-1} \cdot \tau)^2}{(\beta^{-2} + \tau^T \cdot K^{-1} \cdot \tau)}
\]

which is smaller than the MSPE of homBLUP as given in Section 2.3, equation (2.3.9).

We can see that, if the number of elements \( n \) is big, then \( \tau^T K^{-1} \tau \) is a very big element compared with \( \beta^{-2} \), and then the improvement may be negligible. On the other hand, if our mean \( \beta \) is very small (for example after applying a trend removing process like "median polish") we can get a noticeable improvement, and the MSPE will be close to that of Simple Kriging. In (5.1.16) we present the estimated mean value to be a function of its true value. In practice, we will apply an iterative process by which the mean is computed as follows:

\[
(\beta^{-2} + \tau^T K^{-1} \tau) \cdot \hat{\beta} = (\tau^T K^{-1} y)
\]

\[
1 + \tau^T K^{-1} \tau \cdot \hat{\beta}^2 = \tau^T K^{-1} y \cdot \hat{\beta}
\]

\[
\hat{\beta}^2 - \left( \frac{\tau^T K^{-1} y}{\tau^T K^{-1} \tau} \right) \cdot \hat{\beta} + \left( \frac{1}{\tau^T K^{-1} \tau} \right) = 0.
\]
By solving the quadratic equation and using the LESS estimation \( \hat{\beta} \) of the mean \( \beta \) we get:

\[
\hat{\beta} = \frac{\hat{\beta}}{2} \pm \sqrt{\frac{\hat{\beta}^2 - 4\sigma_\beta^2}{4}} = \hat{\beta} \pm \frac{\hat{\beta}}{2} \sqrt{1 - \frac{4\sigma_\beta^2}{\hat{\beta}^2}},
\]

(5.1.19)

It is important to mention that Schaffrin (1997) introduced a Kriging predictor, which is a compromise between the homBLUP and homBLIP, namely the Best Linear \( \sigma_0^2 \)-Softly Unbiased Predictor or the \( \sigma_0^2 \)-BLISUP. The parameter \( \sigma_0^2 \) describes the degree of softness and may vary from zero to infinity. Consequently, this parameter can generate either the classical OK or the Optimal Biased Kriging predictor in the extreme cases; and it reflects the user confidence in the prior information about the mean of the underlying process. A detailed description of the formulas for this predictor and their derivation is given in Schaffrin (1997, 2001). From the discussion above and equations (5.1.16-19) it is clear that in practice the Optimal Biased Kriging should be computing using the homeogram matrix, thus avoiding the unnecessary problem of computing the quadratic estimators of \( \beta \) (which may not even exist; see Schaffrin 2000b). Consequently, in the subsequent sections we use only equations (5.1.13-15). Implementation of those equations requires an inversion of the \( n \times n \) homeogram matrix (\( n \) is the number of given data points). This complex operation is a major obstacle in trying to apply this method to a data set, which is bigger than a few hundreds of points. In the next section we will present a method to alleviate this problem.
5.2 Spatial coherency matrix inversion

5.2.1 Stability of the coherency matrix

A major obstacle of spatial statistics methods is the high computational complexity. This has led to many investigations into improved algorithms and the use of high performance computers; see for example Li (1996) who reports on ways to use parallel processing for spatial statistics algorithms. Equations (5.1.14) explain the main reason for the heavy computational burden in Kriging, namely the need to solve a $n \times n$ linear system, invert and perform multiplication with a $n \times n$ spatial coherency matrix. The practical solution of taking a subset of the given data points to reduce the matrix dimensions has been reviewed in Chapters 3 and 4. However, in the case of Optimal Biased Kriging, this practice is not directly applicable because the biased prediction for each point would cause synthetic and conspicuous discontinuities in each sample point. It is therefore important to find a reliable algorithm to efficiently perform matrix operations with any of the spatial coherency functions. In general, the stability and existence of the inverse matrix is a function of many factors; singularity or non-invertibility of the matrix occurs according to von Frese (1998) if

- a row (or column) contains only zeros;
- two or more rows (or columns) are linearly dependent (a linear combination of each other becomes zero; also known as "collinearity").

A more common condition in practice is that the system is nearly singular, namely if

- the elements of a row (or column) are nearly zeros;
• two or more rows (or columns) are nearly linearly dependent or show a high degree of near collinearity.

Near singularity of the inverse covariance matrix will result in large and erratic prediction values and also large Mean Square Prediction Errors. In that case the system is said to be unstable. One way to assess the degree of near linear dependence is the relative size of the eigenvalues, specifically the reciprocal condition number \( \frac{\lambda_{\text{min}}}{\lambda_{\text{max}}} \)

where \( \lambda_{\text{min}} \) and \( \lambda_{\text{max}} \) are the minimum and maximum eigenvalues of the spatial coherency matrix, respectively. Davis and Morris (1997) study the condition number of the coherency matrix as a function of the data network, the semi-variogram model, the nugget effect, the range and the sill in a series of experiments. They came to the following conclusions:

• An increased sampling density is clearly associated with increasing condition numbers.

• An increased nugget value has a beneficial effect on the condition number.

• Gaussian semi-variogram models can experience more problems than other models.

• Relatively high condition numbers are generally associated with a relatively small average Kriging variance.

In our case study data, the small or near zero eigenvalues are caused by duplicated or nearly duplicated points which do not really contribute independent information. Since we have enough observations, we are able to delete those data points, which are associated with smaller eigenvalues. To detect those points we spatially sorted our data
(see next section for the discussion of the Morton ordering scheme) and deleted points that are very close to each other (relative to the resolution). Of course, reliability will be negatively affected; for more details, see Schaffrin (2000c).

5.2.1 Matrix inversion methods

We will now assume that the matrix is stable and invertible, and consider suitable algorithms to invert the spatial coherency matrix. The problem of inverting a non-singular matrix \( \mathbf{A} \) is central in scientific computations, and a wealth of algorithms has been developed to solve it. A wide range of iterative methods has been proposed (Strang 1988, pp. 380-386) which express the equation \( \mathbf{Ax} = \mathbf{b} \) (or \( \mathbf{AA}^{-1} = \mathbf{I} \)) in an update form

\[
\mathbf{S} \cdot \mathbf{x}_{k+1} = \mathbf{T} \cdot \mathbf{x}_k + \mathbf{b} \tag{5.2.1}
\]

where \( \mathbf{S} \) and \( \mathbf{T} \) are chosen so that \( \mathbf{A} = \mathbf{S} - \mathbf{T} \) and \( \mathbf{S} \) is an easily invertible matrix such as the triangular part of \( \mathbf{A} \) in the Gauss-Seidel method or the diagonal part of \( \mathbf{A} \) in Jacobi's method. Thus, under mild conditions, the sequence \( \mathbf{x}_k \) should converge to the true solution of \( \mathbf{x} \) where:

\[
\mathbf{x}_{k+1} = \mathbf{S}^{-1} (\mathbf{T} \cdot \mathbf{x}_k + \mathbf{b}) \tag{5.2.2}
\]

Iterative methods however may be unstable or even fail to converge to the true solution. Other methods include pivoting and factorization of the matrices to facilitate Gauss' elimination; see, for example, Golub and van Loan (1989, chapters 3 and 5). We will briefly review a very efficient method known as Cholesky's factorization. This method expresses a symmetric positive-definite matrix as the product of a triangular matrix and its transpose:

\[
\mathbf{A} = \mathbf{R}^\mathsf{T} \mathbf{R}
\]
where $R$ is an upper triangular matrix, see also Koch (1999, pp. 30-33). If $A$ is positive-definite, then all the diagonal elements of $A$ are positive and the off-diagonal elements will be "not too big." If the size of $A$ is $n \times n$, the computational complexity of Cholesky's factorization of $A$ is $O(n^3)$, but the complexity of the subsequent Gaussian elimination solutions is only $O(n^2)$.

Davis and Grivet (1984) and Barry and Pace (1997) exploit the sparseness of the covariance matrix to solve the Kriging equations. If an observation displays stochastic dependency with its nearest $m$ neighbors, only $m$ non-zero entries exist per row of the spatial covariance matrix $K$. If we want to apply this method to the matrix $H$, we may express it as:

$$H = \bar{H} + (\tau \cdot \alpha \cdot \tau^T)$$

where $\alpha$ is a constant which is equal to the plateau value of the homeogram and causes $\bar{H}$ to be positive-definite and sparse, with zeros at values above the range. As a result we can compute the inverse of $H$ by:

$$H^{-1} = \bar{H}^{-1} - \bar{H}^{-1} \cdot \tau \cdot (\alpha^{-1} + \tau^T \bar{H}^{-1} \tau)^{-1} \tau^T \bar{H}^{-1}.$$  

In case of the covariance matrix specifically, the matrix $K$ will contain $nm$ non-zero elements out of $n^2$ possible elements. This produces a $m/n$ proportion of non-zero elements (Pace and Barry 1997). Sparsity enhances computational efficiency; for example, multiplying the $n \times n$ matrix $A_1$ by the $n \times n_2$ matrix $A_2$ requires $O(n_2 \cdot n^2)$ operations using dense matrices while with the equivalent sparse matrices multiplication computation requires $O(n_2 \cdot n \cdot m)$ operations. We refer also to the discussion of Press et al. (1992, pp. 157-159) about the efficiency of Strassen's matrix multiplication. The
enhancement is even more significant for matrix inversion. As we saw, this operation can be built upon the Cholesky factorization (R^T R) or, more generally, the LU factorization (lower times upper triangulate matrix) which, when using Gaussian elimination, requires \( O(n^3) \) operations for dense matrices. However, if the matrix is sparse and has a band structure with lower bandwidth \( p \) and upper width \( q \) the operation require \( O(2npq) \) flops; a complete algorithm is given in Golub and van Loan (1989, pp.150-151). Unfortunately, the existence of a pure band structure is not very common.

Example 5.1 demonstrates the problem of constructing a band-structured matrix; 16 data points and their covariance matrix \( K_t \) are displayed in upper part of Figure 5.4 and Table 5.1.1 respectively. Those 16 points are part of the laser data points, and we used the following covariance function to construct the \( K \) matrix:

\[
C(h) = \begin{cases} 
\sigma_y^2 \left[ 1 - 1.5 \left( \frac{h}{r} \right) + 0.5 \left( \frac{h}{r} \right)^3 \right] & 0 < h \leq r \\
0 & h = 0
\end{cases}
\]

which is a spherical anisotropic model with \( \sigma_y^2 = 0.596 \text{ m}^2 \) (sill) and \( r = 42 \text{ m} \) (range). Even with this small area, as we have only 100*100 meters in this example, the covariance matrix \( K \) exhibits strong sparseness with 170 zero elements out of 16*16=256 matrix elements. However, the seemingly unorganized fashion of the zeros is also apparent and, therefore, we should find an efficient method to arrange the covariance matrix before the processing.

5.2.3 Constructing a band-limited structured matrix

Many methods have been proposed to minimize the bandwidth of matrices (or their envelops); unfortunately this problem is known to be NP-complete (see Skiena 1997, pp.128
Thus any solution for this problem relies on the so-called "brute-force" methods or ad-hoc heuristics. Those methods have been extensively studied in the numerical analysis community and increasingly benefit from recent developments in graph theory. In this review we will mention two important methods:


The algorithm first finds a leftmost vertex of the associated graph of the matrix (see Figure 5.1 for an example of a matrix graph). It then generates a level structure by breadth-first search and orders the vertices by decreasing distance from the leftmost vertex. The Gibbs - Poole - Stockmeyer algorithm is another method which is based on the same principles (Meurant 1999, pp. 141-142); empirical evaluation of these and other algorithms on a test suite of 30 matrices have shown that Gibbs - Poole - Stockmeyer appears to be the best method (Skiena 1997, p. 203). However, the worst case of this method still requires $O(n^3)$ operations which would wash out any possible saving for inverting the matrix. All the proposed methods are oblivious of the spatial source of the data, and they have to analyze a bigger data set of size $n \times n$ than the initial data set of $n$ sample points. In this section, we propose a preprocessing step to our data which will enable us to construct a system which is almost sparse from its immediate formation. The proposed preprocessing step consists of spatial sorting and will be performed on the data before the construction of the covariance matrix. It had already been mentioned by Davis and Culhane (1983) that arranging the data by increasing x-coordinate produces a matrix, which has a smaller bandwidth.
Figure 5.1: The covariance matrix and its associated graph; each diagonal point is numbered and is given a vertex in the matrix graph. Now the bandwidth reducing problem is equivalent to renumbering the vertices in the graph (from Snay 1976).

In this work we move one step farther by applying an improved ordering technique. Many ordering techniques have been proposed and adopted; see, for example, Saalfeld (1998a), and Abel and Mark (1990). We have decided to use the "Peano Key", also known as the Morton Ordering technique. This method rearranges the data using both the X-coordinate and the Y-coordinate by interleaving the X-,Y- coordinate numbers as presented in Figures 5.2 and 5.3. Compared to other methods, the Morton Order has the lowest mean absolute difference measures and the overall minimum value of Moran statistics. The Morton Order is both monotonic and quadrant-recursive according to Abel and Mark (1990) who compared a variety of method for spatial sorting. Additionally, Morton Ordering can be performed using a Radix sorting which has a worst case of $O(n \cdot \log n)$ algorithm time (Skiena 1997, pp. 237-238).
We applied the Morton Ordering method by using binary representation of the coordinates (see Figure 5.2) and thus attaining sorting with higher resolution. The lower part of Figures 5.4 presents the data from Example 5.1 after applying Morton Ordering.
and the produced covariance matrix is presented in Table 5.1.2. The improvement using our method is apparent. The covariance vector $\kappa(s')$ may also be sparse when we have a coherency function with a finite range. This can be used in the prediction formula (2.3.12) for $\bar{x}(s')$ and also in the corresponding MSPE formula where the term

$$\kappa(s')^T K^{-1} \kappa(s')$$

can be computed more efficiently by partitioning $K^{-1}$ into $m\times m$ blocks if the bandwidth (non-sparseness) of $\kappa(s')$ is $m$.

5.3 Implementation and experimental results

This section will describe our experiments with Optimal Biased Kriging. We use the laser data set which was described in Section 1.3.2. These data include 1099 laser scanned elevation points and independently collected test data of 331 points. Our first step was to detrend the data using a 3rd order polynomial as described in Section 2.1.3. An important reason to detrend the data before using the homeogram is computational; the large values (because of the additional squared mean - $\beta^2$) of the homeogram matrix elements cause large eigenvalues which are occasionally too close to the finite computational precision of the computer. Figure 5.5 presents the polynomial surface using a 3*3 meter grid. In the experiments we have compared Ordinary Kriging, with homeogram coherency function given in (2.3.21), to Optimal Biased Kriging as given in (5.1.11).

Experiment 5.1: We used the entire subset of 595 points (laser2.txt) and interpolated the test data set of 331 points (lasertest1.txt). The results are given in Table 5.2. The average value of height: 0.02m with a maximum of 3.34m and minimum of -1.35m
Figure 5.4: Upper part, original data points of Example 5.1; the numbers indicate the point location in the file. Lower part, Data points after applying Morton Ordering, the numbers are the assigned file locations.
Table 5.1.1: Original covariance matrix $\mathbf{K}$

\[
\begin{array}{cccccccccccccccc}
0.47 & 0.24 & 0.06 & 0.13 & 0.03 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.24 & 0.47 & 0.23 & 0.02 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.06 & 0.23 & 0.47 & 0.13 & 0.06 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.13 & 0.02 & 0.47 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.03 & 0.1 & 0.13 & 0.47 & 0 & 0 & 0 & 0 & 0.05 & 0.05 & 0.02 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.06 & 0 & 0 & 0.47 & 0.17 & 0.05 & 0.12 & 0 & 0 & 0 & 0 & 0.01 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.17 & 0.47 & 0.23 & 0.04 & 0 & 0 & 0.05 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.05 & 0.23 & 0.47 & 0.04 & 0 & 0 & 0.19 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.02 & 0 & 0.05 & 0.12 & 0.04 & 0.04 & 0.47 & 0 & 0 & 0.09 & 0.01 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.02 & 0.05 & 0 & 0 & 0 & 0 & 0 & 0.47 & 0.08 & 0.09 & 0 & 0.01 & 0 \\
0 & 0 & 0 & 0.02 & 0 & 0 & 0 & 0 & 0 & 0.08 & 0.47 & 0.02 & 0 & 0.29 & 0.07 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.09 & 0.02 & 0.47 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.01 & 0.05 & 0.19 & 0.09 & 0 & 0 & 0 & 0.47 & 0 & 0 & 0.01 \\
0 & 0 & 0 & 0 & 0.01 & 0 & 0 & 0 & 0.01 & 0.01 & 0.29 & 0 & 0 & 0.47 & 0.18 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.07 & 0 & 0 & 0.18 & 0.47 & 0.02 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.01 & 0.02 & 0.47 & 0
\end{array}
\]

Table 5.1.2: Covariance matrix $\mathbf{K}$ computed from data after applying Morton Ordering.
Figure 5.5: Laser data points with the 3rd order polynomial surface. Grid resolution is three meters.
Table 5.2: Results of the comparison between the OBK and OK predictions methods for large data set (see Chapter 4, p. 101, for a definition of the parameters)

<table>
<thead>
<tr>
<th>Method</th>
<th>Actual MSPE (m³)</th>
<th>Mean MSPE/ (m³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal Biased Kriging (Homogram)</td>
<td>0.35759204</td>
<td>6.794577</td>
</tr>
<tr>
<td>Optimal Biased Kriging (Homogram) – trend removed</td>
<td>0.3458</td>
<td>0.1731</td>
</tr>
<tr>
<td>Ordinary Kriging (Homogram)</td>
<td>0.35759411</td>
<td>6.7945800</td>
</tr>
<tr>
<td>Ordinary Kriging (Homogram) – trend removed</td>
<td>0.3459</td>
<td>0.1731</td>
</tr>
</tbody>
</table>

True average of the data that was predicted: -35.37081

Mean of the data as predicted by OK predictor: -35.424217

Difference: 0.05340483

Mean of the data as predicted by OK with detrending predictor: -35.4279

Difference: 0.0571

Mean of the data as predicted by OBK predictor: -35.423891

Difference: 0.053078549

Mean of the data as predicted by OBK predictor: -35.4269

Difference: 0.0560
This result is not a surprise, looking at equations (5.1.17) and (2.3.23) we could see that for large data sets the coefficient $\overline{v}(s')$ becomes very small. However, it is important to notice that in contrast to what is expected from biased Kriging, namely our results should be biased; we got results, which had almost no bias and even less biased then the ordinary Kriging in both the detrended and original data. The key issue in the understanding of this inconsistency is the homeogram. The homeogram carries information about the mean through the $\sigma^2_e$ parameter. Our interpolated points have a mean of -35.37081 and $(-35.37081)^2=1251.0942$ which is close to the homeogram plateau parameter ($\sigma^2_e=1264$ see table 2.2 for the non-detrended data). This also explains the high and inappropriate value for the OBK/OK prediction average variance. An incorrect estimation of $\eta$ even by a few small percentages will lead to a wrong $\eta(0)$ and quite a big error in the MSPE estimation (see Chapter 2 equation 2.3.23 and Chapter 5 equation (5.1.13). In our derivation at Section 5.1 we assumed a known homeogram, which is free of errors; in practice the homeogram is estimated and the effect of this step should be taken care of in the global analysis of this process.

These results also agree with Gotway and Cressie (1993) which perform experiments on 9 points data set and compute relative risk of the James-Stein estimator and the BLUE as a function of the norm of $\beta$ i.e. $(\beta' \beta)^{1/2}$. The greatest reduction in risk of (relative risk of 99.89% which is quite small already) occurs for $\beta$ near zero. As $\beta$ increase the improvement in the relative risk becomes negligible.

Computation of Kriging without using sparse matrices techniques took more then three times when we used sparse matrices techniques.
CPU time for the non-sparse OBKP of data interpolation was: 872.4 seconds.

CPU time for the ordered OBKP data interpolation was: 342.18 seconds

Using the Morton Order sorting technique, we were able to improve our computational speed by an average of 15%, namely:

CPU time for the non-ordered OBKP of data interpolation was: 284.87 seconds.

We assessed the performance of the algorithm using CPU time command, this benchmark should be applied many times to get a meaningful results. On all the tests that we have performed we could see an improvement. Since we believe that evaluating an algorithm with experiments has many limitations see the comments made by Tipper (1995), we have decided not to pursue this method of empirical evaluation of the algorithm and to rely on the analytical analysis as given before.

Experiment 5.2: We compared Ordinary Kriging predictor to the Optimal Biased Kriging predictor (OBK) on small data sets which were already detrended. We cut two data sets from the main data set and computed the interpolation results, which are given in Tables 5.3 and 5.4.

Data set 1: We have 15 points and 8 test points to interpolate:

Average value of the interpolated value (residuals of the height after detrending) = 0.45m

Range of 2.23m; average value of elevation = -35.00875m;

Average value of OBK interpolation plus trend = -34.65625m;

Average value of OK interpolation plus trend = -34.6501m.
### Table 5.3: Results of the comparison between the OBK and OK prediction methods for the small data set no. 1.

<table>
<thead>
<tr>
<th></th>
<th>Optimal Biased Kriging (Homeogram)</th>
<th>Ordinary Kriging (Homeogram)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual MSPE (m^2)</td>
<td>0.241</td>
<td>0.248</td>
</tr>
<tr>
<td>Mean MSPE / MSE (m^2)</td>
<td>0.131</td>
<td>0.1315</td>
</tr>
</tbody>
</table>

**Data set 2:** Our data include 20 points and 12 test points to interpolate:

Average value of the Kriged residuals after detrending = -0.42m;

Range of 3.78m; Average value of elevation = -35.00875m;

Average value of OBK interpolation plus trend = -34.65625m;

Average value of OK interpolation plus trend = -34.6501m.

### Table 5.4: Results of the comparison between the OBK and OK prediction methods for small data set no. 2.

<table>
<thead>
<tr>
<th></th>
<th>Optimal Biased Kriging (Homeogram)</th>
<th>Ordinary Kriging (Homeogram)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual MSPE (m^2)</td>
<td>0.4226</td>
<td>0.4242</td>
</tr>
<tr>
<td>Mean MSPE / MSE (m^2)</td>
<td>0.1395</td>
<td>0.1396</td>
</tr>
</tbody>
</table>
The experimental results show an improvement in the interpolation accuracy of the Optimal Biased Kriging over the Ordinary Kriging. For large data sets the improvement is small whereas for small data sets the decrease in the Mean Squared Prediction Error is recognizable. Hence this procedure should be considered in specific cases of small data sets when accuracy is a critical factor. A typical example for this scenario can be found in Cheng (2001) where very few GPS sea surface measurements need to be interpolated, and with careful examination of the results.
CHAPTER 6

CONCLUSIONS AND FUTURE RESEARCH

This work, as the title suggests, described new methods for spatial statistics in Geographic Information Systems. We have endeavored to detail the practical difficulties which arise in applying the geostatistical technique for massive data sets in GIS and to suggest some ways of overcoming them.

We have described a Delaunay triangulation based neighborhood search method for Kriging interpolation. This method provides a new and faster way to interpolate massive data sets; it is more accurate than other search methods that were described before in the literature. Moreover, we have demonstrated the superiority of this Delaunay structure based Kriging interpolation over the conventional and (in the GIS and Civil Engineering community) quite popular Triangular Irregular Network (TIN) based linear interpolation.

During the development process we have analyzed the computational efficiency of our algorithm and reviewed different methods that were suggested for the implementation of those computational geometry algorithms.

Another spatial prediction method which was thoroughly investigated is the Optimal Biased Kriging. We have reviewed the underlying concepts of this method and derived the required equations. We then developed a new method to overcome the computational problem associated with Optimal Biased Kriging. Basically this method performs spatial
sorting on the data, and consequently the proper matrices become sparse which significantly eases the Kriging computation.

The presented work indicates a need for future research with respect to the following issues:

- An improvement for the computational time of the spatial coherency functions (Section 2.2) can be achieved by spatial sorting (see Section 5.2) and using the maximum distance as a criterion to take only points within this particular neighborhood.
- It is advisable to check for the reliability of the results. In Section (3.5), we proposed a new measure to evaluate the quality of Kriging interpolation, namely by comparing the average MSPE of linear interpolation (often named Triangular Irregular Network based interpolation) to the average MSPE of Kriging interpolation. This comparison is very simple to perform and does not require any extensive computation if the Delaunay structure is already available. Naturally it should be performed on an independent set of observations. Another novel approach which must be considered in this context was developed by Schaffrin (2000c)
- Break-lines and holes need to be included in the Delaunay triangulation structured Kriging. Let P be a set of n points in $\mathbb{R}^2$ and L a set of non-crossing constraint edges with endpoints in P. A point $P_i \in \mathbb{R}^2$ is visible from a site q if the open segment $P_iQ$ does not intersect any edge of L. A constrained Delaunay triangulation (CDT see also Lee and Lin (1986)) of P using L is a triangulation of
P which extends of the edges in L so that the circulmcircle of every triangle contain no site that is visible from all three sites defining the triangle. The extension of this definition to our Kriging algorithm by including holes and boundaries is trivial; and it can be executed using a 2D visibility algorithm in GIS. The extension of break-lines is also immediate; however, it gives rise to some questions regarding the validity of some initial assumptions. For example; stationarity, which we used for the spatial dependence function estimation. A solution for that may be found by dividing the data to a few sub data sets which are separated by break-lines, each data set will have its own spatial dependence function in addition to the spatial dependence function between data sets.

- A combined algorithm, which uses the first order Delaunay neighborhood and first-order Voronoi neighborhood, will have the advantages of both including all the closest neighbors and avoiding extrapolation (as demonstrated in the counter examples of Figures 3.5 and 3.10). This algorithm was not yet fully developed; but, based on the duality of Voronoi diagram and Delaunay triangulation, it should be fairly simple to design an algorithm which has no additional significant computation time.

- This research (Section 4.2) attempts to develop some tool to evaluate the effect of extrapolation on the interpolation accuracy. The methodology presented gave us some indication about the extrapolation error for our specific magnetic data case study. This method can work for any data set given its semi-variogram. However,
it was not yet generalized to the two-dimensional case nor developed into a universal proof of extrapolation Vs. interpolation.

- In Section 4.4.3 we examined the required interpolation neighborhood. A complete theorem that analyzes the distortion of the original signal by using only a subset of it for the interpolation was beyond the scope of our research. Nevertheless, we demonstrated the use of spectral representation to analyze the signal. Since the output of the interpolation process (with subset selection) is a multiplication of the signal with a Rect function (in the spatial domain) and a convolutions in the frequency domain, with a Sinc function, it would be interesting to examine the Modular Transfer Function (MTF) of the interpolation operator with limited neighborhood. The MTF describes the change between input and output signal. (Schenk 1999, p. 26). An investigation of how the level of the interpolation neighborhood affects this function was not performed and may be a worthy topic for future research.

- Today, geographic information technology can automatically identify certain set of parameters for Kriging interpolation (see Gribov et al. 2001). The next generation GIS should be able to automatically identify and choose the proper interpolation method. In this work we proposed to use Delaunay triangulation based linear interpolation as the standard to compare between methods, due to its simplicity and uniqueness (with a few rare exceptions)

- In Section 5.2 we dealt with inverting the covariance/ semi-variogram/ homeogram matrix. A better algorithm should be developed for the inversion of
the covariance matrix; this algorithm may use Green's function to replace the continuous covariance function. More research is needed to analyze degeneracies in the input data that cause a singular covariance/semi-variogram / homeogram matrix (The obvious case is that of duplicate points or the case of the undetrended homeogram matrix.) Solutions for this problem in the case of least-squares adjustment are well established by means of a generalized inverse matrix (See Schaffrin 1999) but should be investigated in the Kriging contest as well.

• In Section 5.2 we proposed Morton Ordering to arrange the covariance / homeogram matrix into a band-limited matrix. It would be interesting to examine the effect of other sorting methods for the arrangement of this matrix.

• In Section 5.3 we used the homeogram as a spatial coherency function within the Kriging equations. More investigations are required to analyze the pros and cons of the homeogram compared to the semi-variogram and covariance function.

The programs and data sets, given in the appendices, provide an efficient way for future research in this direction.
APPENDIX A

DATA SETS AND PROGRAM DESCRIPTIONS

A.1 Data sets and their descriptions

A.1.1 Magnetic data file:

Magdata1.txt includes 597 points of aero-magnetic data which are taken from W/E flight lines, thus creating a grid with approximately 1000 meter horizontal spacing and 2500 meter vertical spacing.

Data format; ASCII with five columns as follow:
Num, X, Y, Corrected_magnetic_field, Trend, Difference

Magtest1.txt includes 95 points of magnetic data along vertical lines to test the efficiency of our algorithm.

Magdata2.txt includes 1322 points of aero-magnetic data that are taken from W/E flight lines with approximately 500 meter horizontal spacing and 2500 meter vertical spacing, and also those from the tie flight lines which have 25 km horizontal separation and 500 meter vertical separation. This data set was also used to derive medium range spatial coherency functions of the magnetic data.

Data format: ASCII with the following columns:
X, Y, Flight_altitude, Corrected_magnetic_field, Bedrock, line_type, Line_number,Trend, Difference
Bedrock column will show 1 when the point is located above rock outcrop.

Line type column will show "stie" when the point is from a tie line.

Trend Column shows the point value from 3rd order polynomial surface.

Difference column shows the difference between the magnetic field and the trend.

Magtest2.txt includes 1322 points that that are taken from W/E flight lines they are between the points of Magdata2.txt i.e. 250 meter from both sides; with approximately 500 meter horizontal spacing and 2500 meter vertical spacing. the tie flight lines points have 25 km horizontal separation and 500 meter vertical separation

Mag11730.txt - part of flight line 11730, used to model the spatial coherency function in the short range distances.

Data format; ASCII with the following columns:

X, Y, Corrected_magnetic_field, Flight_altitude, DEM, Difference, Bedrock

A.1.2 Laser scanning data set:

Laser1.txt - 1099 points of laser scanning data with the following format:

X, Y, Z, Trend, Residual

Trend column shows the point value from 3rd order polynomial surface.

Residual column shows the difference between the real height Z and the trend.

Lasertest1.txt - 331 points of laser scanning data with the following format:

X, Y, Z, Trend, Residual

Trend column is the point value from 3rd order polynomial surface.

Residual column is the difference between the real height Z and the trend.
A.1.3 Ancillary data sets:

Test_s.txt - a data set with 42 points made in order to provisional tests of the programs

Test_sp.txt - 4 points to be predicted using test_s.txt

Format: Num, X, Y, Z

A.2 Matlab programs

A.2.1 Programs used in Chapter 2:

dens_data2.m - Program to read a file of magnetic data and take every 10th point from it and put in a sample file and the 10 point which is 5 place below point it will put in a another file for algorithm validation.

CovarS.m - program to present the experimental covariance function, homeogram, semi-variogram and to prepare an ASCII file that summarizes the value of those files at 30 rows.

The user needs to define the maximum distance, number of lags above 15 to present, and the column of the data parameter. Output format is:

number_pairs average_distance Semi_variogram Covariance Homegram

fitfun.m - program to compute the homeogram or covariance function model using a linearized exponential or Gaussian models. The program will use the output file of program CovarS.m and perform LESS fitting of the function, then it allows the user to interactively make last final changes.

Figure2_2_1.m - This program was used to create Figure 2.2 it contains the principal spatial coherency functions and their implementation.
A.2.2 Programs used in Chapter 3.

TIN8.m - program that controls the Delaunay structured Kriging interpolation. It reads two ASCII files: the sample data and locations to interpolate and output the interpolation results file and the Kriging variance file.

Btriangle1.m - sub-function which is being used by tin8.m to evaluate the neighborhood level.

Striangle.m - sub-function which is being used by tin8.m to evaluate the zero-order neighborhood level.

Triangle.m - sub-function which is being used by tin8.m to perform Delaunay structured linear interpolation.

Krigin.m sub-function which is being used by tin8.m to perform Ordinary Kriging.

Test.m - program that computed a few parameters of the interpolation results for example the MSPE of the actual interpolation results.

Test3.m - program that computed a few parameters from the kriging variance file for example the mean and the max of the MSPE.

A.2.3 Programs used in Chapter 4.

Figure4_7.m program to compare a signal, its spectrum and its covariance function.

figure4_56 - A program that compute the spectrum of the signal; As presented in Figures 5 and 6.

kriging_ext.mws - MAPLE program to compute extrapolation error of Ordinary Kriging.

LESS_EXT.mws - MAPLE program to compute extrapolation error of polynomial fitting.
A.2.4 Programs used in Chapter 5.

**Morton.m** – program that sorts the data following the Morton Ordering scheme.

**OkrigingH.m** - program that computes Ordinary Kriging using the homeogram.

**BlasKH1.m** – program that computes Optimal Biased Kriging using the homeogram.

**BiasK.m** – program that computes Ordinary Kriging using the Semi-variogram.

**Test4.m** – program that computes a few parameters of the interpolation results for example the MSPE of the actual interpolation results on the output files.
APPENDIX B

Mathematical proofs.

B.1 Ordinary Kriging prediction

The derivation of the ordinary Kriging can be found in many places for example Cressie (1993) pp. 119-123 or Armstrong (1998) pp. 88 and Chiles and Delfiner (1999 pp. 164-168). However, in geodesy Kriging formulas are related with least squares collocation and the derivation of kriging has the flavor of adjustment problem more than of prediction problem. We could not find a detailed derivation of the Kriging formulas with this approach and therefore we have added this appendix. We rely mostly on B. Schaffrin and Y. Bock (1994) in the following derivation. They have derived the homBLUP (best homogeneously linear Unbiased estimation) in the general case with the design matrix \( A \neq I \), and not as a prediction problem. In the case of homBLUP \( \beta \) is not given (in contrast with simple Kriging) and we need to compute it. To make this appendix self contained we rewrite the mathematical model of (2.3.1)

Mathematical Model: Our observed process is given by:

\[
y = x + e \\
\tau_\beta = x + e_0
\]

and

\[
\begin{bmatrix}
e \\
e_0
\end{bmatrix} \sim 
\begin{bmatrix}
0 \\
0
\end{bmatrix} 
\begin{bmatrix}
\sigma^2 \cdot I & 0 \\
0 & C_x
\end{bmatrix}
\]

(B.1.0)

where

\( n_y := [y(s_1), ..., y(s_n)]^T \) is the nx1 vector of observed process
\[ n \mathbf{x}_1 := [x(s_1), \ldots, x(s_n)]^T \] is the nx1 random effects vector of actual process value (our prediction is \( x \) at location \( s' \))

\[ \mathbf{e}_i := [e_1, \ldots, e_n]^T \] is the nx1 random observation error

\[ \mathbf{e}_0 := [e_{01}, \ldots, e_{0n}]^T \] is the nx1 random process mis-centering vector

\( \beta_1 \) is the process mean value (usually unknown)

\( n \mathbf{C}_x \) is the n\times n dispersion matrix of \( x \) with the elements \( C_x(s_i, s_j) \)

\( \sigma_c^2 \) is the variance component for the observational noise

\( s \tau_1 = [1, 1, \ldots, 1]^T \) is the nx1 "summation vector"

The three conditions of the homBLUP are:

1. Homogeneously linear
   \[ s \bar{x}_i = s \mathbf{L}_x \cdot y_i \] (B.1.1)

2. Unbiasedness
   \[ \beta = E(\bar{x}) = L \cdot E(y) = L \cdot E(x) = L \cdot \beta \]
   \[ \Rightarrow s \mathbf{L}_x \cdot s \tau = s \tau \]
   where \( \tau = [1, \ldots, 1]^T \) (B.1.2)

3. Best or minimum trace of the MSPE matrix,
   \[ tr \cdot MSPE(\bar{x}) = tr[D(\bar{x}) - C_x(\bar{x}, \bar{x}) - C_x(\bar{x}, x')] = \]
   \[ tr[D(x') - C_x(x', x')]L^T - L \cdot C_x(x + e, x') + LD(x + e)L^T = \] (B.1.3)
   \[ = tr[D(x') - C_x(x', x + e)L^T - L \cdot C_x(x + e, x') + LD(x + e)L^T] \]

MSPE(\( \bar{x} \)) = D(\( \bar{x} - x \)) since the prediction is unbiased, \( E(\bar{x} - x) = 0 \)

There is no correlation between the process and the measurement error:

\( C(x', e) = 0; C(x, e) = 0 \)

If we use the spatial representation and the short notations of::

\[ n \mathbf{x}(s') \_1 := [C_x(x(s_1), x(s')), \ldots, C_x(x(s_n), x(s'))]^T \] (B.1.4)
We assume that the covariance function is isotropic and stationary

\[ C(x,x') = C(x',x) = \begin{bmatrix} C_x(x(s_i),x(s_i)) \end{bmatrix} = k(s') \text{ and} \]

\[ C_x(x(s'), x(s')) = C(0) = \sigma_x^2 = C_x(s', s') \]

\[ \text{tr} \cdot MSPE(\bar{x}) = \text{tr}\{ [C_x(s', s') - \kappa(s') \cdot L^T - L \cdot \kappa(s')^T + L \cdot (\sigma_e^2 \cdot I + C_x) \cdot L^T ] \} = \]

\[ = \text{tr}\{ [C_x(s', s') + L \cdot (\sigma_e^2 \cdot I + C_x) \cdot L^T - 2 \cdot L \cdot \kappa(s')^T ] \} \]  (B.1.5)

Recall that:

\[ D(x) = D(Ly') = L \cdot D(e + x) \cdot L^T = L \cdot (\sigma_e^2 \cdot I + C_x) \cdot L^T \]

\[ C_x(x', x') = C_x(Ly, x') = LC_x^\prime \{(x_s + e_i, x') = LC_x^\prime \{x_s, x'\} + LC_x^\prime \{e_i, x'\} = L \cdot \kappa(s') \]

Thus we write the Lagrange target function as:

\[ \Phi(L^T, v^T) = \text{tr} \{ C_x(s', s') + L(\sigma_e^2 \cdot I + C_x) \cdot L^T - 2 \cdot L \cdot \kappa(s') + 2 v^T (\tau^T \cdot L^T - \tau^T) \} \]  (B.1.6)

The Eular Lagrange necessary conditions (first partial derivatives) are:

\[ \frac{\partial \Phi(L^T, v^T)}{2 \cdot \partial L^T} = (\sigma_e^2 \cdot I + C_x) \cdot L^T - \kappa(s') + \tau \cdot v = 0 \]  (B.1.7)

\[ \frac{\partial \Phi(L^T, v^T)}{2 \cdot \partial v^T} = L \cdot \tau - \tau = 0 \]  (B.1.8)

Multiplying (B.1.7) by \( \tau^T \cdot (\sigma_e^2 \cdot I + C_x)^{-1} \) and using (B.1.8) we get the Lagrange multiplier
\[
-\tau T \cdot L^T + \tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \kappa(s') = \tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \cdot \tau \cdot \nu
\]

\[
\Rightarrow n \nu_1 = \frac{(-\tau T + \tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \kappa(s')))}{\tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \cdot \tau}
\]

Arranging equation (B.1.7) we get:

\[
_n L^T n = (\sigma^2_e \cdot I - C_x)^{-1} \cdot (\kappa(s') - \tau \cdot \nu)
\]

(B.1.10)

Thus we get:

\[
\bar{x} = Ly = [\kappa(s') - \tau \cdot \frac{(-\tau T + \tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \kappa(s'))}{\tau T \cdot (\sigma^2_e \cdot I + C_x)^{-1} \cdot \tau}] \cdot [\sigma^2_e \cdot I - C_x]^{-1} \cdot y
\]

(B.1.11)

Now we will use (B.1.11) to predict values at a single point \(s'\); In this case \(\kappa(s')\) is a \(n^*l\) vector of the covariance between values at location at values on the sample points. Moreover; For Brevity we will arrange (B.1.11) and use a shorter notation also used in Schaffrins (1992,1997,2000)

\[
_n K_n = (I - \sigma^2_e + C_x) \quad \text{the covariance of the observed process } y(s_i)
\]

(B.1.12)

We write the matrices with their row and columns numbers. (B.1.13)

\[
\bar{x}(s') = (1 \cdot \kappa(s'))^T n - \frac{(-\tau T \cdot n \cdot 1 \cdot \tau T \cdot n \cdot K^{-1} n \cdot n \cdot \kappa(s') \cdot 1 \cdot \tau T \cdot n)}{\tau T \cdot K^{-1} \cdot \tau} \cdot n \cdot K^{-1} n \cdot n \cdot y_1
\]

Since the transpose of a number is equal to itself:

\[
\tau T \cdot K^{-1} \cdot \kappa(s') = (\tau T \cdot K^{-1} \cdot \kappa(s'))^T = \kappa(s')^T \cdot K^{-1} \cdot \tau
\]

and thus:
\( \tilde{x}(s') = (\kappa(s')^T - \frac{(-1 + \kappa(s')^T K^{-1} \cdot \tau)}{\tau^T \cdot K^{-1} \cdot \tau} \cdot \tau^T \cdot K^{-1} \cdot \tau) \cdot K^{-1} \cdot y = \) (B.1.14)

\[
= \kappa(s')^T K^{-1} \cdot y + \frac{\tau^T \cdot K^{-1} \cdot \kappa(s')^T - \kappa(s')^T K^{-1} \cdot \tau}{\tau^T \cdot K^{-1} \cdot \tau} \cdot \tau^T \cdot K^{-1} \cdot \tau \cdot y
\]

The BLUE (or LESS) of the mean of \( y(s_i) \) and \( x(s_i) \) is:

\[
\hat{\beta} = \frac{\tau^T \cdot K^{-1} \cdot y}{\tau^T \cdot K^{-1} \cdot \tau}
\] (B.1.15)

We use (A.3.12) to shorten our formulas:

\[
\tilde{x} = \kappa(s')^T \cdot K^{-1} \cdot y - \hat{\beta} + k(s')^T \cdot K^{-1} \hat{\beta}
\]

Or

\[
\tilde{x} = \hat{\beta} + k(s')^T \cdot K^{-1} (y - \hat{\beta})
\] (B.1.16)

The sufficient condition is satisfied since \( K \) is a positive definite matrix.

\[
\frac{\partial \Phi(L^T, \lambda^T)}{2 \cdot \partial L \cdot \partial L^T} = (\sigma^2 \cdot I + C_x) > 0
\] (B.1.17)

We will now derive the MSPE of the ordinary kriging and in order to make our derivation simpler we will develop first the MSPE of the simple Kriging Prediction \( \tilde{x} \).

\[
\tilde{x} = \beta + k(s_0)^T \cdot K^{-1} (y - \tau \cdot \hat{\beta})
\] (B.1.18)

\[
\text{MSPE}(\tilde{x}_s) = D(\tilde{x}_s - x') = [D(x') - C_x (x', \tilde{x}_s) - C_x (\tilde{x}_s, x') + D(\tilde{x}_s)] =
\]

\[
= C_x (x', x') - C_x (x', x + e) L^T - \text{LC}_x (x + e, x') + \text{LC}_x (x + e, x + e) L^T =
\]

If we use the spatial notions of B.1.4

And the simple kriging linear transformation matrix \( L = \kappa(s') \cdot K^{-1} \)
\[ \begin{align*}
\text{MSPE}(\mathbf{x}(s')) &= \sigma_x^2 - \mathbf{k}(s')^T \mathbf{K}^{-1} \mathbf{k}(s') \\
\text{(B.1.19)}
\end{align*} \]

Using that result we will derive the MSPE of the ordinary Kriging: (B.1.20)

\[ \text{MSPE}(\tilde{\mathbf{x}}(s')) = D(\tilde{\mathbf{x}}(s') - \mathbf{\hat{x}}_s(s') + \mathbf{\hat{x}}_s(s') - \mathbf{x}(s')) = \]

\[ = D\{\mathbf{x}(s') - \mathbf{\hat{x}}_s(s')\} + D\{\mathbf{\hat{x}}_s(s') - \mathbf{x}(s')\} - 2\text{COV}\{[\mathbf{x}(s') - \mathbf{\hat{x}}_s(s')],[\mathbf{x}(s') - \mathbf{x}(s')]\} \]

Note: (B.1.20) is true only under the unbiasedness condition.

We will show that the last term in (B.1.20) is zero.

\[ \text{COV}\{[\mathbf{x}(s') - \mathbf{\hat{x}}_s(s')],[\mathbf{x}(s') - \mathbf{x}(s')]\} = \]

\[ \text{COV}\{[(\mathbf{\hat{\beta}} - \mathbf{\kappa}(s')^T \mathbf{K}^{-1} \cdot \mathbf{\tau} \cdot \mathbf{\hat{\beta}} - (\mathbf{\beta} - \mathbf{\kappa}(s')^T \mathbf{K}^{-1} \cdot \mathbf{\tau} \cdot \mathbf{\beta})],[\mathbf{\kappa}(s')^T \mathbf{K}^{-1} \mathbf{y} - \mathbf{x}(s')]\} = \]

The constant shift \( \beta \) has no effect and we plug (B.1.15) and get:

\[ \text{COV}\{[(1 - \mathbf{\kappa}(s')^T \mathbf{K}^{-1} \cdot \mathbf{\tau} \cdot \mathbf{\hat{\beta}}'])[\mathbf{\kappa}(s')^T \mathbf{K}^{-1} \mathbf{y} - \mathbf{x}(s')]\} = \]

\[ (1 - \mathbf{\kappa}(s')^T \mathbf{K}^{-1} \cdot \mathbf{\tau}) \cdot \{\text{COV}\{\mathbf{\hat{\beta}}, [\mathbf{\kappa}(s')^T \mathbf{K}^{-1} \mathbf{y}]\}\} - \text{COV}\{\mathbf{\hat{\beta}}, \mathbf{x}(s')\} = \]

\[ (1 - \mathbf{\kappa}(s_0)^T \mathbf{K}^{-1} \cdot \mathbf{\tau}) \cdot \frac{\{\mathbf{\tau}^T \mathbf{K}^{-1} \cdot \mathbf{\tau}\}}{\mathbf{\kappa}^T \mathbf{K}^{-1} \cdot \mathbf{\tau}} \text{COV}(\mathbf{y}, \mathbf{\kappa}(s')) - \text{COV}(\mathbf{\kappa}(s'), \mathbf{x}(s')) = 0 \]

\[ \Rightarrow \text{COV}\{[\mathbf{x}(s') - \mathbf{\hat{x}}_s(s')],[\mathbf{x}(s') - \mathbf{x}(s')]\} = 0 \quad \text{(B.1.23)} \]

Now we that we continue:

\[ \text{(B.1.24)} \]
$$\text{MSPE}(\tilde{x}(s_i)) = D\{(\tilde{x}(s') - \tilde{x}_s(s'))\} + \text{MSPE}\{\tilde{x}_s(s')\} =$$
$$= D[1 - \kappa(s')^T K^{-1} \cdot \tau \hat{\beta}] + \text{MSPE}\{\tilde{x}_s(s')\} =$$
$$= (1 - \kappa(s')^T K^{-1} \cdot \tau) \cdot \left\{ \frac{(\tau^T K^{-1})^r}{\tau^T K^{-1} \cdot \tau} \cdot \frac{D(y)}{D} \cdot \frac{(1 - \kappa(s')^T K^{-1} \cdot \tau)^r}{\tau^T K^{-1} \cdot \tau} \right\}$$
$$+ \text{MSPE}\{\tilde{x}_s(s')\} =$$
$$= \text{MSPE}\{\tilde{x}_s(s')\} + \frac{(1 - \kappa(s')^T K^{-1} \cdot \tau)^2}{\tau^T K^{-1} \cdot \tau}$$

Note $\kappa(s')^T K^{-1} \cdot \tau$ is a number and thus equal to its transpose.

Finally we get with (B.1.19):

$$\text{MSPE}(\tilde{x}(s')) = \sigma_x^2 - k(s')^T K^{-1} k(s') + \frac{(1 - \kappa(s')^T K^{-1} \cdot \tau)^2}{\tau^T K^{-1} \cdot \tau}$$

(B.1.25)

### B.2 Optimal Biased kriging predictor (homBLIP)

To compare this predictor with the homBLUP, we write our formulas in term of the bias $\beta$ and covariance's function using the of B.1.4, B.1.12 and the prediction formula 5.1.11:

$$\tilde{x}(s') = (k(s') + \beta \beta^T)^T \cdot (K + \beta \beta^T)^{-1} y$$

(B.2.1)

If we use matrix identity from Koch (1998) pp34 - 1.112

$$(A + BD^{-1}C)^{-1} = A^{-1} - A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}$$

(B.2.2)

To rewrite (B.2.1) and get:

$$\tilde{x}(s') = (k(s') + \beta^2)^T \cdot [K^{-1} - K^{-1} \tau (\beta^{-2} + \tau^T K^{-1} \tau)^{-1} \tau^T K^{-1}] y$$

(B.2.3)

We shorten our notation with $\tilde{x}(s') = \tilde{x}$ and multiply the elements to get: (B.2.4)

$$\tilde{x} = (k(s')^T \cdot K^{-1}[y + \tau (\beta^{-2} + \tau^T K^{-1} \tau)^{-1} \tau^T K^{-1} y] + \beta^2 \cdot [1 - \tau^T K^{-1} \tau (\beta^{-2} + \tau^T K^{-1} \tau)^{-1}] \tau$$

157
if we denote \( \hat{\beta} \) by:

\[
\hat{\beta} = \frac{(\tau^T K^{-1} y)}{(\beta^{-2} + \tau^T K^{-1} \tau)^{-1}} \tag{B.2.5}
\]

\[
\Rightarrow (\tau^T K^{-1} y) = (\beta^{-2} + \tau^T K^{-1} \tau) \cdot \hat{\beta} \tag{B.2.6}
\]

and we plug (B.2.5) into (B.2.4):

\[
\tilde{x} = k(s')^T \cdot K^{-1} [y - \tau \hat{\beta}] + \beta^2 \cdot (\beta^{-2} + \tau^T K^{-1} \tau) \hat{\beta} - \beta^2 \cdot \tau^T K^{-1} \tau \cdot \hat{\beta} \tag{B.2.7}
\]

\[
\tilde{x} = k(s')^T \cdot K^{-1} [y - \tau \cdot \hat{\beta}] + \beta^2 \cdot (\beta^{-2} + \tau^T K^{-1} \tau - \tau^T K^{-1} \tau) \cdot \hat{\beta} \tag{B.2.8}
\]

notice that (B.2.8) is identical to (5.1.1) by separating \( y \) we can derive the transformation matrix.

\[
L_{1n} = (\kappa(s')^T - \frac{-\tau^T + \kappa(s')^T \cdot K^{-1} \tau \cdot \tau^T}{\beta^{-2} + \tau^T \cdot K^{-1} \cdot \tau}) \cdot K^{-1} \tag{B.2.9}
\]

The Kriging variance is derived using the simple Kriging MSPE(\( x_s \)) as an initial step

\[
\text{MSPE}(\tilde{x}) = D(\tilde{x} - \bar{x}_s + \bar{x}_s - x) + E(\tilde{x} - x)^T \cdot E(\tilde{x} - x) =
\]

\[
= D(\tilde{x} - \bar{x}_s) + D(\bar{x}_s - x) - 2\text{COV}([\tilde{x} - \bar{x}_s], [\bar{x}_s - x]) + E(L_1 y - x)^T \cdot E(L_1 y - x) =
\]

We have proved in (B.1.21-23) that the COV term is zero the same can be easily applied here to prove that the middle term in (B.2.10) is zero.

\[
\Rightarrow \text{COV}([\tilde{x} - \bar{x}_s], [\bar{x}_s - x]) = 0 \tag{B.2.11}
\]

\[
\text{MSPE}(\tilde{x}(s_i)) = D(\tilde{x} - \bar{x}_s) + \text{MSPE} (\bar{x}_s) + E(\tilde{x} - x)^T \cdot E(\tilde{x} - x) \tag{B.2.12}
\]
Now let's look at the derivation in terms of the Covariance function.

We will derive each of the three terms is (B.2.13) separately:

Right most term of the left side of equation (B.2.12):

\[
D\{(\tilde{x}(s') - \hat{x}(s')) = D[1 - \kappa(s')^T K^{-1} \tau \hat{\beta}] =
\]

\[
= (1 - \kappa(s')^T K^{-1} \tau) \cdot \left\{ \frac{(\tau^T K^{-1})}{\beta^2 + \tau^T K^{-1} \tau} \right\} \cdot D(y) \cdot \left\{ \frac{(\tau^T K^{-1})}{\beta^2 + \tau^T K^{-1} \tau} \right\} \cdot \left\{ \frac{(1 - \kappa(s')^T K^{-1} \tau)^T}{(\beta^2 + \tau^T K^{-1} \tau)^2} \right\}
\]

Left most term of the left side of equation (B.2.12):

\[
E(\tilde{x} - x) = \left\{ \frac{(\kappa(s')^T - (\tau^T K^{-1} \tau)}{\beta^2 + \tau^T K^{-1} \tau} \right\} \cdot E(X) = \left( \frac{(\kappa(s')^T - (\tau^T K^{-1} \tau)}{\beta^2 + \tau^T K^{-1} \tau} \right) \cdot \left( \frac{(1 - \kappa(s')^T K^{-1} \tau)^T}{(\beta^2 + \tau^T K^{-1} \tau)} \right)
\]

We now plug in the values of the above terms in the MSPE equation.
Finally we got with (B.1.19 - the simple Kriging MSPE):

\[
MSE(\tilde{x}(s')) = \frac{(1-\kappa(s'))^T \cdot K^{-1} \cdot \tau^T \cdot K^{-1} \cdot \tau \cdot (1-\kappa(s'))^T \cdot K^{-1} \cdot \tau}{(\beta^{-2} + \tau^T \cdot K^{-1} \cdot \tau)^2} + MSE(\tilde{x}_S(s'))
\]

(B.2.13)

Notice that we can transpose a number:

\[
\tau^T \cdot K^{-1} \cdot k(s') = k(s')^T \cdot K^{-1} \tau
\]
APPENDIX C

FIGURES OF COMPUTATIONAL GEOMETRY

NEIGHBORHOOD PROBLEM

If we preclude incomplete levels due to edge effects, we can define the following neighborhood hierarchy of irregularly distributed point-sets: A zero-order Delaunay neighborhood will have the three points (vertices) which make up the triangle surrounding the point that we want to predict. The first-order Delaunay neighborhood will have 5 to 6 points: three from the zero order triangulation and 2-3 are vertices from the triangles that has a common edge with the previous-order neighborhood (i.e. zero-order Delaunay neighborhood). The second-order Delaunay neighborhood will have a minimum of 7 points and a maximum of 12 points, and so on;
Figure C.1: three levels of triangulations, with different types of points configuration.

<table>
<thead>
<tr>
<th>Type</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Vertices</td>
<td>Triangles</td>
<td>Vertices</td>
</tr>
<tr>
<td>Equilateral triangle</td>
<td>6</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>Right triangle</td>
<td>6</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Isosceles triangle</td>
<td>5</td>
<td>4</td>
<td>7</td>
</tr>
</tbody>
</table>
REFERENCES


and Geodetic Science., The Ohio State University, Columbus, Ohio, 92 pp.


[27] Cressie, N.A.C., 1985a. When are relative variograms useful in Geostatistics? Mathematical Geology, 17(7), 693-702.


165


[120] Saalfeld, A., 1998b. Geodetic Science 786, class notes. Report, Department of Civil and Environmental Engineering and Geodetic Science The Ohio State University, Columbus, Ohio.


Reading, MA, 361-385.


[127] Schaffrin, B., 1999. Geodetic Science 862, class notes. Department of Civil and Environmental Engineering and Geodetic Science The Ohio State University, Columbus, Ohio.


scanners. Report No. 19, Department of Civil and Environmental Engineering and Geodetic Science. The Ohio State University, Columbus.


