BAYESIAN SPATIAL MODELING OF COMPLEX AND HIGH DIMENSIONAL DATA

A Dissertation

by

BLEDAR KONOMI

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2011

Major Subject: Statistics

BAYESIAN SPATIAL MODELING OF COMPLEX AND HIGH DIMENSIONAL DATA

A Dissertation

by

BLEDAR KONOMI

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Bani K. Mallick
	Huiyan Sang
Committee Members,	Jianhua Huang
	Yalchin Efendiev
Head of Department,	Simon J. Sheather

December 2011

Major Subject: Statistics

ABSTRACT

Bayesian Spatial Modeling of Complex and High Dimensional Data.

(December 2011)

Bledar Konomi, B.S., Athens University of Economics and Business

Co-Chairs of Advisory Committee: Dr. Bani K. Mallick
Dr. Huiyan Sang

The main objective of this dissertation is to apply Bayesian modeling to different complex and high-dimensional spatial data sets. I develop Bayesian hierarchical spatial models for both the observed location and the observation variable. Throughout this dissertation I execute the inference of the posterior distributions using Markov chain Monte Carlo by developing computational strategies that can reduce the computational cost.

I start with a "high level" image analysis by modeling the pixels with a Gaussian process and the objects with a marked-point process. The proposed method is an automatic image segmentation and classification procedure which simultaneously detects the boundaries and classifies the objects in the image into one of the predetermined shape families. Next, I move my attention to the piecewise non-stationary Gaussian process models and their computational challenges for very large data sets. I simultaneously model the non-stationarity and reduce the computational cost by using the innovative technique of full-scale approximation. I successfully demonstrate the proposed reduction technique to the Total Ozone Matrix Spectrometer (TOMS) data. Furthermore, I extend the reduction method for the non-stationary Gaussian process models to a dynamic partition of the space by using a modified Treed Gaussian Model. This modification is based on the use of a non-stationary function and the full-scale approximation. The proposed model can deal with piecewise non-

stationary geostatistical data with unknown partitions. Finally, I apply the method to the TOMS data to explore the non-stationary nature of the data.

DEDICATION

To my parents, Kostaq and Urania Konomi, for their unconditional support and sacrifices

ACKNOWLEDGEMENTS

During the course of my Ph.D. studies I have been fortunate enough to benefit from inspiring interactions with people in the statistical community, and I am delighted to be able to acknowledge these here. I would like to express my thanks to my advisor, Dr. Bani Mallick, and my co-advisor, Dr. Huiyan Sang, who guided me through my research and whose comments often sparked my deeper interest in directions which I would have ignored otherwise. Dr. Mallick with his advice on how to approach the problems and Huiyan with her close collaboration, the long discussions helped me sort out the technical details of my work and gave me the spark to develop new ideas.

I would also like to express my thanks to all of the professors that I had in my graduate studies at Texas A&M University as well as my undergraduate studies at Athens University of Economics and Business who not only taught me statistics but also shaped me as a scientist. Additionally, I am very grateful to have worked with other students, such as Soma Dhavala, in my research problems.

Finally, and most importantly, I am sure that none of this would have been possible without the love and patience of my family who have always been there when I needed. Dedicating this dissertation work to them is the minimum I can do to express my thanks.

TABLE OF CONTENTS

Page

ABSTRACT iii

DEDICATION v

ACKNOWLEDGEMENTS vi

TABLE OF CONTENTS vii

LIST OF TABLES ix

LIST OF FIGURES x

CHAPTER

I INTRODUCTION 1

II OBJECT CLASSIFICATION OF GOLD NANOPARTICLES USING BAYESIAN MARKED POINT PROCESS MODEL . . 5

2.1 Nanoparticles in an image and the use of MPP 7
2.2 Data 11
2.3 Object specification 12
2.4 Templates used in the current study 14
2.5 Model 19
2.6 Posterior computation using MCMC 22
2.7 Simulations 30
2.8 Application to gold nano particle analysis 34
2.9 Concluding remarks 42

III REDUCED-DIMENSION HIERARCHICAL STATISTICAL MODELS 44

3.1 Gaussian process models for spatial data sets. 47
3.2 Reduction models 55
3.3 Bayesian inference with predetermined subregions . . 59
3.4 Implementation 62
3.5 Simulations 64
3.6 Data 71
3.7 Concluding remarks 78

CHAPTER Page

IV MODIFIED TREED GAUSSIAN PROCESS 80
4.1 Bayesian inference with undefined dynamic subregions 81
4.2 Bayesian treed Gaussian process 82
4.3 Modified proposed BTGP for TOMS data 83
4.4 Reduction methods in the proposed BTGP 86
4.5 Spatial prediction 87
4.6 Simulation study 88
4.7 Real data analysis 94
4.8 Concluding remarks 97
V CONCLUSION 98
REFERENCES 100
APPENDIX A 108
VITA 121

LIST OF TABLES

TABLE Page

1 MAP estimates of the parameters for the first six objects in Ex1 . . . 35
2 MAP estimates of the parameters for the first six objects in Ex2 . . . 37
3 MAP estimates of the parameters for the first six objects in Ex3 . . . 39
4 Posterior estimations of the model parameters and the MSPE 70
5 MSPE table 76
6 Posterior estimation and the MSPE for two subregions 92
7 MSPE for four different methods 93

LIST OF FIGURES

FIGURE		Page
1	Example of TEM images	8
2	Templates with at least one random pure parameter	17
3	Simulation of two different images with $m = 10$ and their corresponding values for the interaction parameter γ_2	31
4	Distribution of the number of objects, m, for two different values of γ_2: (a) $\gamma_2 = 0$ and (b) γ_2 is considered random	32
5	The last 4000 simulated values of γ_2 for the two different cases: (a) $\gamma_2 = 40$ and (b) $\gamma_2 = 10$	33
6	Simulation of objects at (a) 1000 iteration (b) 1500 iteration. Except from the different movements a merge and a change template move has occurred	33
7	Simulation of objects at (a) 1400 iteration (b) 1700 iteration. Except from the different movements a split template move has ocured	34
8	Object shapes sampled using MCMC in Ex1	36
9	Object shapes sampled using MCMC in Ex2	38
10	Object shapes sampled using MCMC in Ex3	40
11	Objects identified by ImageJ in Ex1. Out of the 22 particles, 4 are recognized. Recognition rate = 18.18%	41
12	Objects identified by ImageJ in Ex2. Out of the 19 particles, 6 are recognized. Recognition rate = 35.58%	41
13	Spatial location of the simulated data and the 40 locations left to produce the MSPE	66
14	MSPE for four different covariance structures in the first simulation	67

FIGURE Page

15 MSPE for four different covariance structures in the second simulation 68

16 Difference of the absolute value of the residuals for two different methods 69

17 Comparing the MSPE for the three different methods 75

18 Comparing the MSPE of anisotropic with isotropic covariance 78

19 Simulated data and 400 knots uniformly distributed 88

20 MCMC posterior distribution of the number of subregions when we use the full model 90

21 MCMC posterior distribution of the number of subregions: a) predictive process with 100 knots b) predictive process with 400 knots c) full-scale with 100 knots and tapering 10 d) full-scale with 400 knots and tapering 10 90

22 MCMC posterior distribution of the number of subregions: a) uniform prior and b) Poission prior 92

23 Histogram of the MCMC values of ζ when the number of subregions is two 94

24 Six different partitions in the tree process 95

25 MAP estimation of the Bayesian treed GP 96

26 Posterior distribution of the parameters using full-scale approximation 96

A-1 Distribution of the MAP estimates for some shape parameters in Ex1 108

A-2 Distribution of the MAP estimates for some shape parameters in Ex2 109

A-3 Distribution of the MAP estimates for some shape parameters in Ex3 110

A-4 Posterior distribution of the parameters using full model for 50 different bands in the latitude range [-70,70] 111

FIGURE Page

A-5 Posterior distribution of the parameters using the predictive process for different band in the latitude range [-70,70] using 250 knots . 112

A-6 Posterior distribution of the parameters using full rank approximation for different bands in the latitude range of [-70,70] with 250 knots and 30 subregions 113

A-7 Posterior distribution of the parameters using the full model for different bands in the latitude range of [-55,55] 114

A-8 Predictive proccess boxplot for different bands in the rage of latitude [-55,55] using 250 knots 115

A-9 Posterior distribution of the parameters using full-scale approximation with 250 knots and 30 subregions 116

A-10 The posterior distribution of the parameter of the "Range" matrix in the 3D model 117

A-11 The posterior distribution of the variance parameter in the 3D model 118

A-12 Level 2 and Level 3 TOMS data 119

A-13 Level 2 and Level 3 TOMS data for the US 120

CHAPTER I

INTRODUCTION

In the last two decades the hierarchical Bayesian methods have been the tool of choice for many scientists to model challenging data sets. The success of the hierarchical Bayesian modeling relay on the ability to realistically model data sets and on the computational efficiency which came with the use of Markov chain Monte Carlo (MCMC). One area that has benefited from the use of the Bayesian modeling is spatial statistics.

Spatial statistics is essential for modeling the heterogeneity and the interaction in environmental, geophysical, image and other spatial data sets. The spatial statistical methods can be divided into three main categories: spatial point process, geostatistics and lattice. In this dissertation, we concentrate our attention on spatial point process and geostatistics. In the spatial point process, the spatial positions (locations) are modeled as random events. On the other hand, in geostatistics the spatial locations are considered to be a continuous over space and the dependence among the responses $Y(s)$ at different locations s are modeled. The Bayesian methods simplify the modeling of these data sets and the parameters can be realistically interpreted. Moreover, the high-dimensional nature of the parametric space and the data sets makes it computationally more attractive than the classical statistics.

The main topic of this dissertation is to apply the hierarchical Bayesian methods in complex and huge data sets. We first model the data realistically and then apply

---

This dissertation follows the style of Journal of Statistical Planning & Inference.

different techniques to reduce the computational cost. We begin in Chapter II by using a marked-point process to model objects in an image. Next in Chapter III we use approximation techniques to deal with non-stationary and high-dimensional data sets. Finally we dynamically model the non-stationary with a Gaussian Tree Model.

Chapter II deals with the Bayesian modeling of the locations of nanoparticles in an image and the classification into a predetermined family of shapes. The properties of materials synthesized with nanoparticles are highly correlated to the sizes and shapes of the nanoparticles. By controlling the shape and size of nanoparticles during synthesis, one could control the properties of the synthesized material. Accurate methodologies for enabling morphological analysis are highly underdeveloped in the current practice of nanomaterial science and engineering. Transmission electron microscopy (TEM) imaging technique can be used to measure the morphological characteristics of nanoparticles, which can be simple circles or more complex irregular polygons with varying degree of scales and sizes. A major difficulty in analyzing the TEM images is the overlapping of objects, having different morphological properties with no specific information about the number of objects present. Furthermore, the objects lying along the boundary render automated image analysis much more difficult. To overcome these challenges, we propose a Bayesian method based on the marked-point process representation of the objects. We derive models, both for the marks which parameterize the morphological aspects and the points which determine the location of the objects, to greatly reduce the complexity of the problem. The proposed model is an automatic image segmentation and classification procedure, which simultaneously detects the boundaries and classifies the nanoparticles into one of the predetermined shape families. We execute the inference by sampling the posterior distribution using Markov chain Monte Carlo (MCMC) since the posterior is doubly intractable. We also designed split and merge moves in addition to spatial birth and

death moves to efficiently infer the number of objects and their shapes. We apply our novel method to several TEM imaging samples of gold nanoparticles, producing the needed statistical characterization of their morphology.

Chapter III deals with the reduction methods in a huge dimension and non-stationary spatial random field. Gaussian process models have been widely used in spatial statistics but face tremendous computational challenges for very large data sets. The model fitting and spatial prediction of such models typically require $O(n^3)$ operations for a data set of size n. Various approximations of the covariance function have been introduced to reduce the computational cost. The predictive process, tapering and lately the full rank approximation are among the most popular approximation techniques used in the recent statistical literature to deal with the computational cost of large spatial data set. All of these techniques are successfully applied in stationary spatial process models where the data are reasonably considered stationary. The non-stationary spatial process have been considered in the predictive process but only in a simulation study with known partition and parameters. Total Ozone Matrix Spectrometer (TOMS) data are proven to be, by previous work, an example of a large spatial data set with non-stationary covariance function. The goal of this chapter is to generalize the full scale approximation into non-stationary processes, apply it to TOMS data and compare it with other approximation techniques. The method used in this chapter is based on the partition of the spatial region into subregions with stationary random fields which can be linked in a non-stationary unique random field.

Chapter IV is an extension of the third chapter which explore the non-stationary of the huge data sets by using a random partition. We propose a model which separates the space with non-stationary covariance function, with a modified Treed Gaussian Model, into subregions with stationary and anisotropic covariance functions. The

modification is a combination of the use of the reduced covariance function proposed by Sang and Huang (2011) and a use of the non-stationary covariance function, proposed by Paciorek and Schervish (2006). The non-stationary covariance function proposed by Paciorek and Schervish (2006) links different reduced stationary covariance functions from the separate subregions into a unique covariance function. This model is applied to TOMS data where as we explain in Chapter III the non-stationarity is coming though the latitude and the prediction performance show good fit. With this method not only the computational cost is reduced but also the non-stationarity is taken into account.

CHAPTER II

OBJECT CLASSIFICATION OF GOLD NANOPARTICLES USING BAYESIAN MARKED POINT PROCESS MODEL

A marked-point process is a random collection of objects falling in some space. The way this objects are distributed over space is usually govern by some physical properties which can be expressed in a mathematical model. The mathematical model is not straightforward to be expressed when these collections of objects appear in an image, since we need a model for the pixels. This difficulty can be resolver with the hierarchical Bayesian model where we can use different model for the pixels as well as for the object interaction and link them.

A mathematical model should be considered for the case where two objects may not overlap completely, but only in a small regions. This behavior can be modeled using a repulsive interaction prior in the point process representation. Since there may be a difference in the degree (intensity) of overlapping from image to image, we assume that the parameters of the point process are unknown and ought to be inferred. This leads to a hierarchical model setting where the prior distribution has an intractable normalizing constant. As a result, the posterior is doubly intractable and we use the Markov-chain Monte-carlo (MCMC) framework to carry-out the inference. Simulating from distributions with doubly intractable normalizing constants has received much attention in the recent literature, but most of these methods consider the normalizing constant in the likelihood and not in the hierarchical prior; Møller et al. (2006) and Murray et al. (2006), Liang (2010), among others. In this chapter, we borrow the idea of Liang and Jin (2011), which is a modified version of the *reweighting mixtures* given in Chen and Shao (1998) and Geyer and Møller (1994), which

can deal with doubly intractable normalizing constants in the hierarchical prior as well. The MCMC algorithm used can be described as a two step MCMC algorithm. We first sample the parameters from the pseudo posterior distribution - which is a part of the posterior that does not contain the AIPP normalizing constant - and then an additional Monte Carlo Metropolis-Hasting (MCMH) step that accounts for this normalizing constant.

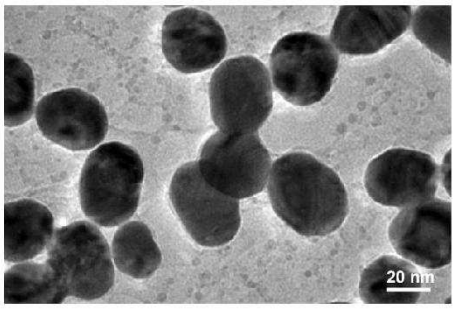
Sampling from the pseudo posterior distribution is also quite challenging. Inferring the unknown number of objects is a complex task. We propose Reversible Jumps MCMC (RJ-MCMC) type of moves to handle both the tasks (Green (1995)). Specifically, we use spatial birth and death moves to sample the *number of objects* and while proposing a new object, we use information from prior knowledge available. However, at times, an object may have to be split into two objects or may have to be merged to form a single object. The regular birth and death moves may be slow in mixing, or we may travel through some very low-probability intermediate states. Due to this specific problem, split and merge moves have been designed. We also propose RJ-MCMC moves to swap (switch) the shape of an object. Using the above mentioned computational scheme, we obtain the posterior distributions for all the parameters which characterize the nanoparticles: number, shape, size, center, rotation, mean intensity, etc.. Owing to the model specification and the computational engine for inferring the model parameters, our approach extracts the morphological information of nanoparticles, detects nanoparticles laying on the boundaries, quantifies uncertainty in shape classification, and successfully deals with the object overlapping, when most of the existing shape analysis methods fail.

2.1 Nanoparticles in an image and the use of MPP

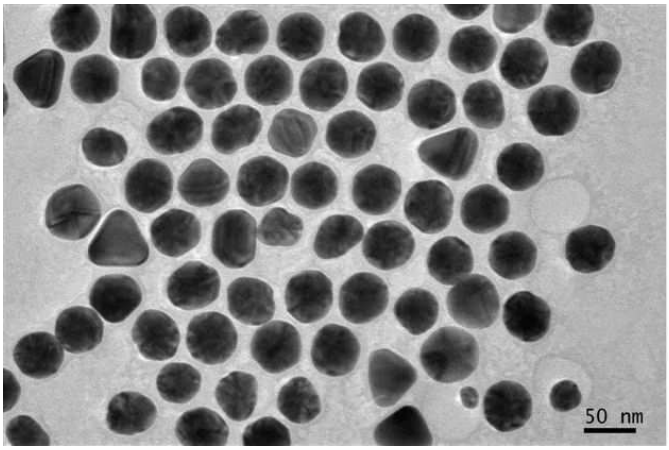
For nanomaterials, material properties are often encoded in their morphological characteristics. Understanding the material properties can help scientists to find new applications and optimize their process parameters. Such knowledge offers control over the morphological footprints Wang et al. (1998); Mohamed et al. (2000); El-Sayed (2001); Nehl et al. (2006); Pan et al. (2007). A major step in accomplishing this task is the availability of image analysis tools that segment, classify and characterize the images of the nanoparticles.

Transmission Electron Microscopy (TEM) has been used to analyze the morphology of nanoparticles. TEM uses a beam of electrons transmitted through an ultra-thin specimen, interacting with the specimen as it passes through. A grayscale image is then formed from the interaction of the electrons with the specimen, which conveys information about the presence or absence of a nanoparticle in two dimensions. In the presence of a nanoparticle, electrons have a difficulty to pass through, resulting in reduced intensity in that part of the image. Representative TEM images are shown in Fig. 1.

Detecting nanoparticles and determining their properties is usually based on the grayscale contrast in the TEM images. High-level statistical image analysis techniques model an image as a collection of discrete objects and are used for object recognition, Baddeley and van Lieshout (1993). Here, the object is any abstract representation of an ensemble of pixels that are of importance to the subject expert. In images with object overlapping, Bayesian approaches have been preferred over maximum likelihood estimators (MLE). The unrestricted MLE approaches tend to contain clusters of identical objects, whereas the Bayesian approaches mitigate this problem by penalizing the overlapping by an area interaction process prior (AIPP) or by a similar



(a) Gold nanoparticles at 20nm



(b) Gold nano particles at 50nm

Figure 1: Example of TEM images

approach, offering flexibility over controlling the overlapping or the touching. Moreover, the high dimensionality of the problem makes the Bayesian approach attractive in these cases. The Bayesian approach has been used quite successfully in a wide variety of settings (Baddeley and van Lieshout (1993); Mardia et al. (1997); Rue and Syversveen (1998); Rue and Hurn (1999); Al-Awadhi et al. (2004a); van Lieshout (2008)), among others.

In Mardia et al. (1997), an area interaction model which forbids objects to overlap completely is proposed. Inference is carried out by finding the Maximum

A Posteriori (MAP) estimates and the area interaction parameters are chosen by simulation experience, in effect, fixing the parameters that define the penalty terms. Spatial birth and death moves are considered to infer the number of objects. There are a number of limitations of this model: 1) we may only find a local mode, 2) when more than one type of object is present, we may need additional transdimensional moves to navigate the rugged posterior energy landscape and 3) fixing certain parameters in the AIPP are very restrictive as it needs intervention by the subject expert every time. Further, their application is limited to circular objects with greater intensity in the center.

Rue and Hurn (1999) also used MPP as in Mardia et al. (1997) to handle the unknown number of objects but introduce polygonal templates to model the objects. However, their application is restricted to cell detection problems, where the objects do not overlap but barely touch each other and the method works more like a segmentation technique than as a classification technique. Moreover, the success of this approach depends on MPP parameters, which are assumed known throughout the simulation. Recently, Al-Awadhi et al. (2004a) used the same model except that they considered elliptical templates instead of polygonal templates and applied their method to similar cell images. As noted, all the above methods take advantage of the MPP, in particular the AIPP or any other prior that penalizes the overlapping or touching. In addition to the above methods, substantial work in estimating the closed contours of objects in an image has been done by Blake and Yuille (1992); Helterbrand et al. (1994); Qian and Mardia (1995); Pievatolo and Green (1998); Hobolth et al. (2002); Jung et al. (2008); Kothari et al. (2009), among others. Imaging processing tools, especially for cell segmentation also exist; for instance, ImageJ (ImageJ (2004)) is a tool recommended by the National Institute of Health (NIH). However, the features of the data we are dealing with are quite different from those considered

in the literature reviewed, as there are various degrees of overlapping of the nanoparticles differing in shapes and sizes, as well as a significant number of nanoparticles lying along the image boundaries. Moreover, in this chapter we are interested in simultaneous classification and segmentation. Efficient object representation is critical to overcome these challenges.

There are various possible statistical approaches for object representation in high-level image analysis. The pattern theory Grenander (1993); Grenander and Miller (1995) suggests using multiple-graph deformable templates and jump-diffusion simulation. The method is based on the deformation of a template - a polygon with a fixed number of sides of variable length - to find the optimal fit to the object. The geometrical object process proposed by Baddeley and van Lieshout (1993) uses marks to handle the varying dimensionality of the geometrical objects and the point process to handle the locations. Rue and Hurn (1999) successfully combine these two methods and generalize the approach to polygonal deformable templates with random number of sides, retaining the marked point process characteristics. Their method uses a polygon template of varying resolution for different shapes, which works well in detecting object boundaries. However, the nanoparticles have usually smooth corners and if we use this method then the number of sides in the polygon will be overestimated. And as a result, it leads to incorrect classification.

In Mardia et al. (1997), objects were represented by templates which constitute the marks and were employed even to detect occlusions. More specifically, a template is a predetermined shape represented with fixed vertices, which can be shifted, scaled, rotated and deformed to represent an object in the image. The line segment joining each vertex to some fixed internal point lies entirely within the polygon. The deformation considered in their work can be seen more as a "small scale deformation" of the boundaries and shape classification can be done only if we know exactly the

parameters of a particular shape: e.g. ellipse with certain parameters. Since the structure of the data we are analyzing is different from literature, we adapt object representation strategies discussed above to the problem at hand. Firstly, when we refer to a shape, we refer to a family of geometrical objects which share certain features; for example, an isosceles and a right triangle both belong to the triangle family. There are five types of possible shapes of the nanoparticles in our problem. The scientific reason is that the final shape of the particle is dominated by the potential energy and the growth kinetics. There is a balance between surface energy and bulk energy once a nucleus is formed. The arrangement of atoms in a crystal determines those energies such that only one of these specified shapes can be formed. We use similar scientific reasons to construct shape templates. These templates are determined by the parameters which vary from shape to shape.

The Bayesian approach proposed above is used and comparison with ImageJ (recommended by NIH) has been conducted to demonstrate the efficiency of our proposed method.

The rest of the chapter is organized as follows: Section 2.3 describes the TEM images, Section 2.4 deals with the object specification procedure, Section 2.5 describes the model specification, Section 2.6 describes the MCMC algorithm, Section 2.7 describes a simulation study and Section 2.8 applies the method to the real data. Conclusions are presented in Section 2.9.

2.2 Data

We analyze a mixture of gold nanoparticles in a H_2O solution. In order to analyze the morphological characteristics, nanoparticles are sampled from this solution onto a very thin layer of carbon film. After the water evaporates, the two dimensional morphology of nanoparticles measured using an Electron microscopy such as TEM.

In our case, a JEOL 2010 high resolution TEM operating at 200-kV accelerating voltage was used, which has 0.27 nm of point resolution. The TEM shoots a beam of electrons onto the materials embedded with nanoparticles and captures the electron wave interference by using a detector on the other side of the material specimen resulting in an image. The electrons cannot penetrate through the nanoparticles, resulting in a darker area in that part of the image. The output from this application will be a eight bit gray scale image where darker parts indicate the presence of a nanoparticle. The gray scale intensity is varying as an integer between 1 and 256. Refer to Figure 1 for examples of TEM images.

Due to the absorption of electrons by the gold atoms, the regions occupied by the nanoparticles look darker in the image. The darkness pattern may vary according to specific arrangements of the atoms inside any single nanoparticle. Additionally, one can see many tiny dark dots in the background, which are uniformly distributed throughout the image region. These dark dots are generated because the carbon atoms of the carbon film also absorb electrons. One may also notice a white thin aura wrapping around the whole or partial boundary of a particle. This is the result of having surfactants on the rim of the particles. The surfactants are added to keep the particles from aggregating in the process of making colloidal gold. Analyzing the shapes of the nanoparticles in a TEM image is primarily based modeling them as objects, whose shapes are parametrized. Treating a nanoparticle as an object is the critical component of our modeling framework, which we discuss in the next section.

2.3 Object specification

An object is specified in a series of steps that allow us to model a wide variety of shapes. They are:

1. Template

2. Shift, Scale and Rotate operators

3. Object multiplicity

We discuss each of them in detail below.

2.3.1 Template

A template is a predetermined shape which is defined by a set of parameters which we call *pure shape parameters* or simply *pure parameters*. We will call the template T a *pure object* and we will specify a pure object by its *pure* parameters as $g_T^0 = \{g_T^0(1), \ldots, g_T^0(q)\}$, where q is the number of parameters, and it varies from shape to shape. For example, a circle with unit radius at the origin $(0, 0)$ can be regarded as a template for circular objects. Likewise, an equilateral triangle with unit sides, centered at the origin with the median aligned to the x-axis can be a template for triangular objects. We can potentially differentiate an equilateral triangle from an isosceles triangle even when they both belong to the triangle family. However, to avoid defining an infinite number of templates, we consider all types of a particular shape to be members of the same template. For example, all types of triangles, such as equilateral, right-angled, etc., are considered to be members of the triangle template. As such, when we refer to a template in this chapter we refer to a family of shapes that has certain characteristics. A family of shapes is formed by deforming some of the pure parameters $\{g_T^0(1), \ldots, g_T^0(q)\}$ in the shape definition. We distinguish g_T^0 parameters as random (unknown) g_T^r, and constant (known) g_T^{co}. The random pure parameters g_T^r cannot be determined exactly by the template or by other components of g_T^0. These random pure parameters affect the overall shape, size and other geometric properties, thereby causing a large scale deformation of the template. These parameters are closely related to the template but for simplicity we ignore the indicator T and use

the notation $g_T^0 = g^0 = (g^r, g^{co})$. The pure parameters are chosen such that the defined template will have an area equal to the area of a unit circle, that is π square units. A template can be shifted, rotated and scaled, still belonging to the same shape family.

We also specify landmarks $l^0 = l^0(1), \ldots, l^0(M)$ as the M equally spaced boundary points of a given template. These landmarks can be determined if one knows the pure parameters. The landmarks will help us representing the shape of the real image. In polar coordinates, these landmarks can be represented as:

$$l^0(k) = c_{0,0} + s^0(k)(\cos(\theta(k)), \sin(\theta(k)))^T$$

where $s^0(k)$ is the distance of k^{th} landmark from the center $c_{0,0}$, and $\theta(k)$ is the rotation of the k^{th} landmark with respect to the baseline. The particular choice of the coordinate system in which the landmarks are represented does not affect the results. Hence, we have chosen to use polar coordinates for the simplicity of the mathematical analysis. We chose ninety landmark points for all the shapes. Simply speaking, these landmarks in an image form the shape. The random deformation of these landmarks results in *small scale* deformation of the template. In this chapter, we focus our attention on the large scale deformation since the main goal is to determine the shape and not making boundary detection or contour tracking, where small-scale deformations are important. Templates used in the current study are given in the next section.

2.4 Templates used in the current study

Using the expert knowledge as stated in section (2.1), the possible shapes of the gold nanoparticles are: *rectangle, circle, ellipse, triangle* and *polygon*. From now on, we will denote this categorical random variable by T. Below we will give the details

of the corresponding template and their related pure parameters.

- Circle: A circle is defined by the equation

$$(u - c_x)^2 + (v - c_y) = s^2$$

where (c_x, c_y) determines the center and s the radius. In this dissertation, we take $(c_x, c_y) = (0, 0)$ and $s = 1$. The pure template is a unit circle centered at $(0, 0)$. This way $g^0(1) = c_x = 0, g^0(2) = c_y = 0$ and $g^0(3) = s = 1$ and no deformation is needed.

- Ellipse: With respect to the usual coordinate axes, the ellipse is described by the equation:

$$\left(\frac{(u - c_x)\cos(\theta) - (v - c_y)\sin(\theta)}{E_1}\right)^2 + \left(\frac{(u - c_x)\sin(\theta) + (v - c_y)\cos(\theta)}{E_2}\right)^2 = 1. \tag{2.1}$$

The pure object template here is the ellipse with center $(c_x, c_y) = (0, 0)$, rotation $\theta = 0$ and the largest distance $E_1 > 1$ and the shortest E_2. Values of $g^r = E_1 \approx 1$ implies that the elliptical shape is closer to the circular shape whereas large values of E_1 indicate a departure from the circular shape. A circle can be treated as an ellipse and the variable $g^r = E_1$ is the measure of deviation of the ellipse from the circle. A threshold to distinguish ellipse from circle in our applications is chosen as $E_1 \in (1.12, 1.4)$ and $E_2 = 1/E_1$ which constrain the area of ellipse to be equal to π. Here, we have one variable $g^0(3) = E_1$ that controls $g^0(4) = E_2$ and two variables that remain constant as $(g^0(1), g^0(2)) = (c_x, c_y) = (0, 0)$. That way, we have $g^{co} = (g^0(1), g^0(2), g^0(4))$ and $g^r = g^0(3)$.

- Triangle: Many approaches can be seen to define the pure template for the triangle. The *Bookstein coordinates* or *Kendall's spherical coordinates* can be

used to define every possible triangle in $\Re^2$, (Dryden and Mardia (1998)). All the images we analyzed are isosceles triangles without sharp vertices. The degree of sharpness (smoothness) of these vertices varies from triangle to triangle. We construct an appropriate template, namely an isosceles triangle with unknown smoothness for its vertices.

We define the template for every isosceles triangle with a height h_1 and half of the unequal side as h_2 (see Fig. 2). To have an area equal to π, we constrain $h_2 = \pi/h_1$. One of the equal angles θ_1 of the isosceles triangle is considered to vary between $[20^o, 90^o]$. After some algebra we have: $h_1 \in \left\{\sqrt{\pi \tan(20^0)}, \sqrt{\pi \tan(90^0)}\right\}$. Sharp edges in the triangle are smoothed by applying a smoothing spline to approximate the edges after a random cutting of the edges. In addition, we can generalize the triangular template to cover all possible triangles, by introducing h_2 as another random quantity and apply the necessary changes in size and center of template. The template with center at the centroids of the smoothed isosceles triangle and the rotation $\theta = 0$, which is the angle of the h_2 with the baseline, will be our template for a triangle (as in Fig. 2).

- Square & Rectangle: The square of sides $\sqrt{\pi}$ and center $(c_x, c_y) = (0, 0)$ will be the template for the square shapes. For the rectangle we need two parameters A_1, A_2 which determine the half length of sides of the rectangle , with A_1 corresponding to the larger side and A_2 corresponding to the smaller side. For details see Fig. 2. A small portion of the vertices are also randomly removed to cover deformed squares and rectangles. To cover all the possible rectangles in our application, we choose $A_1 \in (A_{11}, A_{12})$ and $A_2 = \pi/4A_1$ which constrain the area of rectangle to be equal to π.

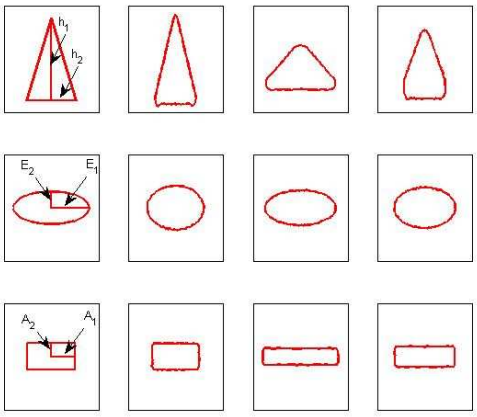


Figure 2: Templates with at least one random pure parameter

2.4.1 Shift, scale and rotate operators

Apart from the parameters that determine the shape which vary from template to template there are also some common parameters related to shifting, rotating and scaling which are needed to represent the actual shape in the image. A particular affine shape with shift $c = (c_x, c_y)$, scale s and rotation θ is given by the landmarks $l = \{l(1), \ldots, l(M)\}$, whose polar coordinates are: $[l(k) = c + c_0 + sS^0(k)(\cos\{\theta(k) + \theta\}, \sin\{\theta(k) + \theta\})^T]$ for $k = 1, \ldots, M$.

2.4.2 Object multiplicity and the Markov point process

In an image, we have multiple objects with different shapes and we assume that the number of objects is unknown. A point process is used to model the unknown number of objects and the overlapping. One of the widely used models that penalize object overlapping is the Markov point process (MPP). Among the many MPPs that control the interaction among the objects in different fashions is the area interaction process prior (Baddeley and van Lieshout (1993)). We calibrated priors such that inference is invariant to changes in the image resolution. The location parameters

$\boldsymbol{c} = (c_1, \cdots, c_m)$ and the number of objects m are modeled as:

$$\pi(\boldsymbol{c}, m|\mathbf{g^r}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T}, \gamma_1, \gamma_2) = \frac{1}{A^*} \exp\left\{-\gamma_1 m - \gamma_2 S(\boldsymbol{\eta})\right\}, \tag{2.2}$$

where $S(\boldsymbol{\eta})$ denotes the area of the image covered by more than one object, $\eta_i = (c_i, s_i, t_i, \theta_i, g_i^r)$ is a collection of parameters that represents the i^{th} object and $\boldsymbol{\eta} = \boldsymbol{\eta}_m = \{\eta_k\}_k^m$ represent these parameters for all objects which we call 'object parameters'. Moreover, $\mathbf{T} = \mathbf{T}_m\{t_i\}_{i=1}^m$, $\mathbf{s} = \mathbf{s}_m = \{s_i\}_{i=1}^m$, $\boldsymbol{\theta} = \boldsymbol{\theta}_m = \{\theta_i\}_{i=1}^m$, $\mathbf{g^r} = \mathbf{g^r}_m = \{g_i^r\}_{i=1}^m$ and A^* is the normalizing constant which depends on all the paprameters described above $(\boldsymbol{\eta}, m)$ and the positive unknown parameters γ_1 and γ_2, $(A^* = A(\boldsymbol{\eta}, m, \gamma_1, \gamma_2))$. The interaction parameter γ_2 controls the overlapping between objects and γ_1 the number of objects in the image. For example, $\gamma_2 = 0$ does not penalize overlapping, whereas $\gamma_2 = \infty$ does not allow overlapping at all. Prior distributions for γ_1 and γ_2 are considered in subsequent sections. For simplicity we introduce $\boldsymbol{\gamma} = (\gamma_1, \gamma_2)$ to represent the MPP parameters.

Another way to penalize object overlapping is the two-way interaction:

$$\begin{aligned}\pi(\boldsymbol{c}, m|\boldsymbol{\eta}) &= \frac{1}{A^*} \exp\left\{-\gamma_1 m - \gamma_2 \sum_{i<j} |R(\eta_i) \cap R(\eta_j)|\right\} \\ &\quad I[\text{No three or more objects have common area}].\end{aligned}$$

The term $I[\text{No three or more objects have common area}]$ will not allow three or more objects to overlap in the same area, $R(\eta_i)$ is the region of a single object characterized by its parameters η_i and $R(\eta_i) \bigcap R(\eta_j)$ is the overlapping area between the i^{th} and the j^{th} object. We can generalize this case to allow more objects to overlap in a region and also penalize with a different parameter γ_k. Investigating such models is out of the scope of the current work.

2.5 Model

2.5.1 The likelihood function

Due to the electron absorption, the mean intensity of the background is larger than the mean intensity of the regions occupied by the nanoparticles. Furthermore, since each nanoparticle has different volume size, the mean pixel intensity for each nanoparticle is different, which is evident from the representative TEM images of gold nanoparticles shown in Fig. 1. It can also be observed that the overlapping regions have usually lower intensity because they absorb more electrons in that region. For tractability, we consider the darkest region to be the dominant region in determining the configuration of the objects with which it is overlapping. Due to specific arrangements of the atoms inside any single nanoparticle the neighboring pixels have similar intensities. An appropriate choice for the covariance function in such scenarios is the popular Conditional Autoregressive (CAR) model, Cressie (1993). Computationally a much simpler model is the independent noise model, (Baddeley and van Lieshout (1993); Mardia et al. (1997); Rue and Hurn (1999).

After analyzing both real and simulated datasets, the posterior specification of the parameters did not change much even if we replaced the CAR model with the independent Gaussian noise model. We denote $\boldsymbol{\mu} = \boldsymbol{\mu}_m = (\mu_0, \ldots, \mu_m)$ as the mean vector and $\boldsymbol{\sigma}^2 = \boldsymbol{\sigma}^2_m = (\sigma^2_0, \sigma^2_1, \ldots, \sigma^2_m)$ as the variance vector for the background and objects intensity. To facilitate the notation, we use $\boldsymbol{\Theta} = (\boldsymbol{\eta}, m, \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$. In this case the likelihood can be written as:

$$f(Y|\boldsymbol{\Theta}) \propto \prod_{p=1}^{N} \exp\left\{ -\frac{1}{2\phi(x_p)} (y_p - \delta(x_p))^2 \right\} \tag{2.3}$$

where N is the number of pixels, x_p is the p^{th} pixel, $\delta(x_p)$ is the mean of the p^{th} pixel, $\phi(x_p)$ is the function of the variance depending on the pixel. More explicitly the mean

intensity for pixels covered by more than one object is taken to be the minimum mean intensity of the objects covering the pixels and with variance which corresponds to the variance of that object.

For example, in the case where we allow only two way interaction, Eqn. (2) can be written as:

$$f(Y|\boldsymbol{\Theta}) \propto \exp\Bigg\{-\frac{1}{2\sigma_0^2}\sum_{\nu\in R(\eta_0)}\big(y_{\nu_0}-\mu_0\big)^2-\sum_{i=1}^{m}\frac{1}{2\sigma_i^2}\sum_{\nu\in R(\eta_i)\setminus R_{(-i)}}\big(y_{\nu_i}-\mu_i\big)^2 \\ -\sum_{i<j}\frac{1}{2\min_{(\mu_i,\mu_j)}(\sigma_i^2,\sigma_j^2)}\sum_{\nu\in(R(\eta_i)\bigcap R(\eta_j))}\big(y_{\nu_{i,j}}-\min(\mu_i,\mu_j)\big)^2\Bigg\} \tag{2.4}$$

where $R_{(-i)}$ is the region occupied by all objects (nanoparticles) without the i^{th} object and $R(\eta_0)$ is the area of the background.

2.5.2 Prior specification

We elicit the joint prior distribution hierarchically as follows:

$$\begin{aligned}\pi(\boldsymbol{\Theta},\boldsymbol{\gamma}) &= \pi(\boldsymbol{\Theta}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma})\\ &= \pi(\boldsymbol{\mu},\boldsymbol{\sigma}^2)\pi(\boldsymbol{\eta},m|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma})\\ &= \pi(\boldsymbol{\mu},\boldsymbol{\sigma}^2)\pi(\boldsymbol{c},m|\boldsymbol{\gamma},\mathbf{g^r},\mathbf{s},\boldsymbol{\theta},\mathbf{T})\pi(\mathbf{g^r},\mathbf{s},\boldsymbol{\theta},\mathbf{T})\pi(\boldsymbol{\gamma}).\end{aligned} \tag{2.5}$$

In the above expression $\pi(\boldsymbol{\mu},\boldsymbol{\sigma}^2)$ is the prior of the means and the variances of the background and the objects, $\pi(\boldsymbol{c},m|\boldsymbol{\gamma},\mathbf{g^r},\mathbf{s},\boldsymbol{\theta},\mathbf{T})$ is the joint prior of the locations and the number of the objects as given in equation (1), $\pi(\mathbf{g^r},\mathbf{s},\boldsymbol{\theta},\mathbf{T})$ is the joint prior on all the 'object parameters' except the locations and $\pi(\boldsymbol{\gamma})$ is the prior on the interaction parameters.

We assume independent (μ_i,σ_i^2) pairs and assign a non-informative prior for each of these pairs.

$$\pi(\boldsymbol{\mu},\boldsymbol{\sigma}^2)=\prod_{i=0}^{m}\pi(\mu_i,\sigma_i^2)\propto\prod_{i=0}^{m}(\sigma_i^2)^{-1}. \tag{2.6}$$

All the 'object parameters' except the locations are assumed to be independent from object to object. Also the scale, rotation and template within object parameters are assumed to be independent of other parameters while g_i^r is assumed to be closely related to the template T_i (shape). We remind the reader that g_i^r are different from template to template. In mathematical form we have:

$$\pi(\mathbf{g^r}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T}) = \prod_{i=1}^{m} \pi(s_i)\pi(\theta_i)\pi(g_i^r|T_i)\pi(T_i). \tag{2.7}$$

We assign a uniform prior for s_i which is proportional to the size of the image $S_{\max}$, i.e., $\pi(s_i) \sim U(0, S_{\max})$. Except for circles, all other shapes have a rotation parameter $\theta \in (0, \pi]$. The prior density for θ is $\pi(\theta) \sim \{|\cos(\theta)| + \pi^{-1}\}/3$, which favors values near $\theta = 0$ and $\theta = \pi$. The circle and square do not have a random pure parameter while the other considered templates have at least one random pure parameter. All these parameters have one basic characteristic: they are constrained to take values between two variables (a_1, a_2). We use altered location and scale Beta distribution as prior given by:

$$\pi(g_i^r) = \frac{1}{Beta(\alpha, \beta)} \frac{(g_i^r - a)^{\alpha-1}(b - g_i^r)^{\beta-1}}{(b-a)^{\alpha+\beta-1}}$$

where a, b, α, β are different for the three different cases. Furthermore, we have used the uniform discrete distribution to specify the prior for the template, T_i.

For both the object process parameters γ_1, γ_2 we assume independent log-normal distribution priors with parameters which determine a mean close to 100 and large variance, $\gamma_1 \sim LN(\alpha_1, \delta_1)$, $\gamma_2 \sim LN(\alpha_2, \delta_2)$.

2.5.3 The posterior distribution

The model proposed above is a hierarchical model of the form:

$$\begin{cases} y|\boldsymbol{\Theta} \sim f(y|\boldsymbol{\Theta}) & \text{(a)} \\ \boldsymbol{\Theta}|\boldsymbol{\gamma} \sim \pi(\boldsymbol{\Theta}|\boldsymbol{\gamma}) \\ \qquad \equiv \dfrac{1}{A^*}\pi^*(\boldsymbol{c}, m|\boldsymbol{\gamma}, \mathbf{g}^{\mathbf{r}}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T})\pi(\mathbf{g}^{\mathbf{r}}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T}|m) & \text{(b)} \\ \boldsymbol{\gamma}|\alpha_1, \delta_1, \alpha_2, \delta_2 \sim \pi(\boldsymbol{\gamma}|\alpha_1, \delta_1, \alpha_2, \delta_2) & \text{(c)} \end{cases} \tag{2.8}$$

where $\alpha_1, \delta_1, \alpha_2, \delta_2$ are known values and $\pi^*(\boldsymbol{c}, m|\boldsymbol{\gamma}, \mathbf{g}^{\mathbf{r}}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T})$ is the MPP prior without the normalizing constant.

The posterior distribution of the parameters $p(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}, m, \boldsymbol{\gamma}|y)$ is proportional to the multiplication of (a), (b) and (c) in the above hierarchical representation.

$$\begin{aligned} p(\boldsymbol{\Theta}, \boldsymbol{\gamma}|y) &\propto \pi(\boldsymbol{\gamma})\pi(\boldsymbol{\mu}, \boldsymbol{\sigma}^2|\boldsymbol{\eta})\pi(\boldsymbol{\eta}|\boldsymbol{\gamma})f(y|\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) \\ &= \frac{1}{A^*}\pi^*(\boldsymbol{c}, m|\boldsymbol{\gamma}, \mathbf{g}^{\mathbf{r}}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T})\pi(\mathbf{g}^{\mathbf{r}}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T})\pi(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)\pi(\boldsymbol{\gamma})f(y|\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) \\ &= \frac{1}{A^*}p^*(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}, m, \boldsymbol{\gamma}|y), \end{aligned} \tag{2.9}$$

which is not only intractable but also has a random intractable normalizing constant A^*. We use Markov-chain Monte-carlo (MCMC) computation algorithm to carry-out the inference since the posterior distribution is analytically intractable and the point process prior has a random intractable normalizing constant. To facilitate the discussion, we call $p^*(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}, m, \boldsymbol{\gamma}|y)$ the pseudo posterior distribution.

2.6 Posterior computation using MCMC

The MCMC algorithm used in this chapter can be described as a two stage Metropolis-Hasting algorithm. We first sample the parameters from the pseudo posterior distribution followed by a Monte Carlo Metropolis-Hasting step to account for A^* (Liang and Jin (2011)).

The MCMC algorithm will have the following form:

- Given the current state $\boldsymbol{\Theta}^k, \boldsymbol{\gamma}^k$ draw $\boldsymbol{\Theta}', \boldsymbol{\gamma}'$ from p^* using any standard MCMC sampler.

- Simulate auxiliary variables $z_1, \ldots, z_M$ from $z \sim f(z; \boldsymbol{\Theta}')$ using an exact sampler.

- Estimate $R = A(\boldsymbol{\eta}', m', \boldsymbol{\gamma}')/A(\boldsymbol{\eta}^k, m^k, \boldsymbol{\gamma}^k)$ as

$$\hat{R} = \frac{1}{M} \sum_{i=1}^{M} \frac{f(z; \boldsymbol{\Theta}')}{f(z; \boldsymbol{\Theta}^k)}.$$

- Compute (estimate) the MH rejection ratio α as $\hat{\alpha} = 1/\hat{R}$.

- Accept $\boldsymbol{\Theta}', \boldsymbol{\gamma}'$ with probability $\min(1; \hat{\alpha})$.

Simulating auxiliary variables z_i from the likelihood is straightforward. The challenge lies in drawing from the pseudo posterior.

A generalized Metropolis-within-Gibbs sampling with a reversible jump step is used to simulate from the pseudo posterior distribution with known number of objects. Additionally, a reversible jump MCMC (RJ-MCMC) with spatial birth-death as well as merge-split move is invoked to sample the number of objects and their corresponding parameters.

We draw from the joint pseudo posterior $p^*(\boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\eta}, \boldsymbol{\gamma}, m|y)$ by alternately drawing from the conditional pseudo posteriors of $\boldsymbol{\mu}, \boldsymbol{\sigma}^2\boldsymbol{\eta}|m, y, \boldsymbol{\gamma}$, $\boldsymbol{\gamma}|\boldsymbol{\mu}, \boldsymbol{\sigma}^2\boldsymbol{\eta}, m, y$ and $m|\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\gamma}, y$, as follows:

- Draw $\boldsymbol{\eta}^{k+1}, \boldsymbol{\mu}^{k+1}, \boldsymbol{\sigma}^{k+1}$ from $p^*(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}|m^k, \boldsymbol{\gamma}^k, y)$ using a Metropolis-within-Gibbs sampler.

- Draw m^{k+1} from the pseudo posterior $p^*(m|\boldsymbol{\mu}^{k+1}, \boldsymbol{\sigma}^{k+1}, \boldsymbol{\eta}^{k+1}, \boldsymbol{\gamma}^k, y)$ using a RJ-MCMC.

- Draw $\gamma_1^{(k+1)}, \gamma_2^{(k+1)}$ from the distribution $p^*(\boldsymbol{\gamma}|y, \boldsymbol{\Theta})$ using an M-H step.

We explain these steps in detail, in the following paragraphs.

2.6.1 Updating $\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}$, given m and $\boldsymbol{\gamma}$

The conditional distribution of $p^*(\boldsymbol{\eta}|\boldsymbol{\mu}, \boldsymbol{\sigma}^2, m, y)$ does not have any closed form and the same is true for the conditional distribution of every component or group of components of $\boldsymbol{\eta}$. A Gibbs sampling step which contains Metropolis-Hasting steps and RJ-MCMC step is utilized. The Metropolis-Hasting updates for $(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma}/\mathbf{T})$ and $\mathbf{T}$ are given next.

2.6.1.1 Metropolis-Hasting updates of $(\boldsymbol{\eta}, \boldsymbol{\mu}, \boldsymbol{\sigma})$ exluding $\mathbf{T}$

Updating $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$: The conditional distribution of $p^*(\mu_j, \sigma_j|.)$ is proportional to the multiplication of (9) with (7): $p^*(\mu_j, \sigma_j|.) \propto \pi(\mu_j, \sigma_j^2) f(y|\boldsymbol{\Theta})$.

Metropolis-Hasting step is used to draw from this posterior with proposal distribution

$$\sigma_j^{-N_j-2} \exp\left\{-\frac{1}{2\sigma_j^2}[(N_i-1)s_j^2 + n(\bar{y}_{j.} - \mu_j)^2]\right\},$$

where $s_j^2 = \frac{1}{N_j-1}\sum_{i=1}^{N_j}(y_i - \bar{y}_j)$, $N_j \in (R_j)$ is the total number of pixels in the region of the proposed shape and $\bar{y}_{j.}$ is the sample mean intensity of the j^{th} object. To draw from this proposal we first draw $\sigma_j^2|. \equiv Inv\chi^2(N_j - 1, \sigma^2)$ and then from $\mu_j|\sigma_j^2, . \equiv N(\bar{y}_{j.}, \sigma_j^2/N_j)$.

Updating s, $\boldsymbol{c}$, $\boldsymbol{\theta}$ and $\mathbf{g^r}$: Metropolis-Hasting step is used to draw from the pseudo conditional posterior distributions of the components of $\mathbf{s}$, $\boldsymbol{c}$, $\boldsymbol{\theta}$ and $\mathbf{g^r}$. To implement the M-H step, we need the proposal distribution which will generate the parameters. These proposals should have some good properties in order for the chain to mix well. The preprocessing is not only helpful to determine the starting values of some of the parameters but also their proposal distribution. More specifically, the

proposal distributions of scaling s_j and location c_j are determined by the preprocessing.

Scaling s_j: As we have already mentioned, all the shapes have a scaling parameter. The choice of templates to have an equal area reduces the number of scaling parameters to one per object. Also it has been chosen such a way that different shapes with same scaling parameter s will have the same area. This is a very important property and its benefits will become obvious when we move from template to template in the MCMC algorithm.

Given the current sample of scaling s_j^k we use $q(s_j, s_j^k) \equiv N(s_j^k, \sigma_{S_j}^2)$ as the proposal distribution for s_j, where $\sigma_{S_j}^2$ is derived from the estimated scale s_j^0 from the preprocessing. In this dissertation we chose $\sigma_{S_j}^2 = s_j^0/10$. Then we use a standard M-H algorithm to draw s_j.

Location c_j: When the number of objects is m, we have $2m$ location parameters (m in the x-coordinates and m in the y-coordinates). Given the current sample of location c_j^k we use $q(c_j, c_j^k) \equiv MN(c_j^k, \sigma_{C_j}^2 I)$ as the proposal distribution, where $\sigma_{C_j}^2$ is the variance for both (x, y) coordinates.

Rotation θ_j: The rotation parameter is present in every template except for the circle. We have used the prior proposal in $\theta_j \in [0, \pi]$.

Random pure parameter g_j^r: An independence sampler with the prior distribution as the proposal, $q(g_j^r, (g^r)^k) = q(g_j^r) = \pi(g_j^r)$ has been used. The sampling has been performed as follows:

1. Generate $(g_j^r)^*$ from $q(g_j^r)$

2. Compute

$$\alpha = \frac{p^*\{(g_j^r)^*, \mu^{k+1}, (\sigma^2)^{k+1}, T^k, s^{k+1}, c^{k+1}, \theta^{k+1}, (g^r)_{1:(j-1)}^{k+1}, (g^r)_{(j+1):m}^k | y\}}{p^*\{(g_j^r)^k, \mu^{k+1}, (\sigma^2)^{k+1}, T^k, s^{k+1}, c^{k+1}, \theta^{k+1}, (g^r)_{1:(j-1)}^{k+1}, (g^r)_{(j+1):m}^k | y\}} \frac{q\{s_j^k, s_j^*\}}{q\{s_j^*, s_j^k\}} \tag{2.10}$$

3. Set $(g_j^r)^{k+1} = (g_j^r)^*$ with probability $min\{1, \alpha\}$ and $(g_j^r)^{k+1} = (g_j^r)^k$ with the remaining probability.

2.6.1.2 Updating the template T_j (swap move)

We can view the problem of shape selection as a problem of model selection between $M_{j,t_1}, \ldots, M_{j,t_D}$, where M_{j,t_i} represents the model with template t_i. Moving from shape to shape is considered a difficult task since not only the pure parameters that characterize the template are different, but also the parameter specification may not have the same meaning across templates. For example, one can argue that the scaling parameter of a circle can be different from the scaling parameter of a triangle. The move from shape to shape is based on the rule that both shapes should have the same area and the centers of both shapes are the same. This increases the likelihood of generating good proposals. For the particular shapes we deal with, the equality of area also means equality of the scaling parameter. This means that all of the above models M_{j,t_i} have the same scaling s_j and location c_j parameters. The rotation parameter, θ, can be chosen such that the proposed shape overlap "matches" as much as possible to the existing shape given the same (s_j, c_j) or simply one may retain the same θ while changing shapes. The 'pure random' parameters are the only parameters that do not have a physical meaning when we change the shape and also their number could vary from shape to shape. Reversible Jump MCMC is used successfully for problems with different dimensionality and is characterized by introducing auxiliary variables for the unmatched parameters (Green 1995). This is the approach we follow here. Two new variables $(u_{T_j} = g^r_{T_j}, v_{T_j} = g^r_{T_j})$ are introduced to make it clear that the pure parameters have different meaning from template to template. For all the shapes, we provide a general algorithm: Let $\psi_j^k = (T_j^k, s_{T_j^k}, c_{T_j^k}, \theta_{T_j^k}, u_{T_j^k})$ denote the current state and $\psi_j^* = (T_j^*, s_{T_j^*}, c_{T_j^*}, \theta_{T_j^*}, v_{T_j^*})$ the proposed state for ψ^{k+1}. The notations of

the parameters are different from the previous sections to show the dependence of the parameters on the model T_j^*, (or template). If $T_j^k \neq T_j^*$, generate $v_{T_j^k}$ from the prior distribution of the v_{T_j} and consider a bijection:

$$(s_{T_j^*}, c_{T_j^*}, \theta_{T_j^*}, u_{T_j^*}, v_{T_j^*}) = (s_{T_j^k}, c_{T_j^k}, \theta_{T_j^k}, u_{T_j^k}, v_{T_j^k}).$$

This bijection is formed from the steps described above. From this bijection it is clear that the Jacobian is equal to identity matrix, $J = I$, and $|J| = 1$.

In summary, the RJ-MCMC algorithm is:

- Select model $M_{T_j^*}$ with probability $q(T_j, T_j^k) = \pi(T_j)$.
- Generate $v_{T_j^k}$ from $\pi(v_{T_j})$
- Set $(s_{T_j^*}, c_{T_j^*}, \theta_{T_j^*}, u_{T_j^*}, v_{T_j^*}) = (s_{T_j^k}, c_{T_j^k}, \theta_{T_j^k}, u_{T_j^k}, v_{T_j^k})$.
- Compute the M-H ratio:
$$\alpha = \frac{p^*(s_{T_j^*}, c_{T_j^*}, \theta_{T_j^*}, v_{T_j^*}|y)\pi(T_j^k)}{p^*(s_{T_j^k}, c_{T_j^k}, \theta_{T_j^k}, u_{T_j^k}|y)\pi(T_j^*)} \frac{\pi(u_{T_j^*})}{\pi(v_{T_j^k})}|J|$$
where J is the Jacobian.
- Set $\psi^{t+1} = (T_j^*, s_{T_j^*}, c_{T_j^*}, \theta_{T_j^*}, v_{T_j^*})$ with probability $\min(1, \alpha)$ and $\psi^{t+1} = (T_j^k, s_{T_j^k}, c_{T_j^k}, \theta_{T_j^k}, u_{T_j^k})$ with remaining probability.

2.6.2 Updating m

Two different types of moves are considered in updating the number of objects: birth-death and split-merge. In the death step, one chosen-at-random object is deleted and in the birth step, one object with parameters generated from the priors is added. In the merge step we consider the case where two objects die and give birth to a new one and in the split step two new objects are created in the place of one. Let $Pr(birth)$, $Pr(death)$, $Pr(split)$ and $Pr(merge)$ be the probabilities of proposing a birth, death, split or a merge move, respectively.

2.6.2.1 Birth and death movement

The spatial birth and death moves are described in Geyer and Møller (1994) and in Geyer and Thompson (1995). The acceptance ratio for these move type of moves is determined by a RJ-MCMC involving a change in the dimension.

In the birth step a new object η_{m+1} is proposed with a randomly assigned center. In this step we increase the dimension of the parameters by Q_{m+1}, all the parameters which describe the proposed object $(\eta_{m+1}, \mu_{m+1}, \sigma^2_{m+1})$. All these new parameters are sampled from the prior distributions of the Q_{m+1} parameters. The introduction of these kind of auxiliary variables leads again to a Jacobian equal to 1 and the M-H ratio is:

$$\min\Big\{1, \frac{p^*(\eta_{m+1}, \mu_{m+1}, \sigma^2_{m+1}, \boldsymbol{\eta_m}, \boldsymbol{\mu_m}, \boldsymbol{\sigma^2_m}|y)}{p^*(\boldsymbol{\eta_m}, \boldsymbol{\mu_m}, \boldsymbol{\sigma^2_m}|y)\pi(\eta_{m+1}, \mu_{m+1}, \sigma^2_{m+1})} \frac{q((m+1)\to m)}{q(m\to(m+1))}\Big\}. \tag{2.11}$$

The death proposal chooses one object, η_j, at random and removes it from the configuration. The M-H ratio for this move is similar to Eqn. (9).

2.6.2.2 Split and merge movement

The details for the split and merge move are more complicated than the move types described above. First we restrict our attention only to the case where we merge two neighboring objects or split one object into two neighbors. The distance between the two neighbors can be approximated by a function of their individual size. This is the approach we take to propose two new objects in the split step. As in the **swap move** described in sec. (5.1.1), when we move from one state to another, we require that the proposed objects have equal area with the existing. In order for the Markov Chain to be reversible we should ensure that every jump step can be reversed. It is important to mention that we can improve the acceptance rate of this move with different proposed algorithms, e.g. Al-Awadhi et al. (2004b), but that is beyond the scope of this work.

To facilitate the representation we will denote by bold characters $\boldsymbol{\eta}$, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ the current state in every move and $\boldsymbol{\eta}_{-(.)}$, $\boldsymbol{\mu}_{-(.)}$ and $\boldsymbol{\sigma}^2_{-(.)}$ the current state values without the (.) objects.

Merge Step: Let's suppose we have two objects and that their parameters are $(\eta_i, \eta_j, \mu_i, \mu_j, \sigma_i^2, \sigma_j^2)$. In the merge step, we move to a new object with parameters $(\eta_h, \mu_h, \sigma_h^2) = (x_h, y_h, s_h, \theta_h, T_h, g_h^r, \mu_h, \sigma_h)$. The equation which link the sizes of the old objects (s_i, s_j) with the new is $s_h = \sqrt{s_i^2 + s_j^2}$. Also x_h and y_h are chosen to represent the "weighted middle" point taking in account the size of each object as $(x_h, y_h) = ((s_j x_j + s_i x_i)/(s_i + s_j), (s_j y_j + s_i y_i)/(s_i + s_j))$. All the other parameters are chosen from one of the "parent" objects or at random.

In order to match the two dimensions, we introduce six auxiliary variables, $(u_1, u_2, u_3, u_4, u_5, u_6)$, which not only would enable us to move from state to state but also are interpretable: $u_1 = \sqrt{(y_j - y_i)^2 + (x_j - x_i)^2}$ is expressing the distance between two centers of the neighboring objects,

$$u_2 = \arctan\left(\frac{(y_j - y_i)}{(\sqrt{(y_j - y_i)^2 + (x_j - x_i)^2})}\right),$$

is the angle created from the union of the two centers (c_1, c_2), $u_3 = (s_i^2 - s_j^2)/(s_i^2 + s_j^2)$ is chosen such that $R_i = R_h\sqrt{(1+u)/2}$ and $R_j = R_h\sqrt{(1-u)/2}$, $u_4 = \theta_2$,$u_5 = T_2$, $u_6 = g_2^2$.

The acceptance ratio, α, in this case is the minimum of one and:

$$\frac{p^*(\eta_h, \mu_h, \sigma_h^2, \boldsymbol{\eta}_{-(i,j)}, \boldsymbol{\mu}_{-(i,j)}, \boldsymbol{\sigma}^2_{-(i,j)}|y)}{p^*(\eta_{(i,j)}, \mu_{(i,j)}, \sigma^2_{(i,j)}, \boldsymbol{\eta}_{-(i,j)}, \boldsymbol{\mu}_{-(i,j)}, \boldsymbol{\sigma}^2_{-(i,j)}|y)} \frac{q(1 \to 2)}{q(2 \to 1)} \frac{\prod_{i=1}^6 \pi(u_i)}{1} |J|, \tag{2.12}$$

where $|J|$ is the determinant of the Jacobian for the transformation and $q(1 \to 2)$ is the split proposed probability and $q(2 \to 1)$ is the merge proposed probability.

Split Step: In the split step, we move from (x,y,s,θ,T,g^r,u_1,u_2,u_3,u_4,u_5,u_6) to (x_1,y_1,x_2,y_2,s_1,s_2,θ_1,θ_2,T_1,T_2,g_1^r,g_2^r). In order to make this move possible, we introduce

six proposal distributions for the auxiliary variables. We propose $u_1/2$ from the prior of the size parameter, u_2 from the prior of rotation parameter, u_3 from $Unif(-1,1)$, u_4, u_5, u_6 from the priors of θ, T and g^r respectively. In order for this move to be reversible we again use the same transform that was used in the merge step.

The acceptance ratio, α, in this case is:

$$\min\left\{1, \frac{p^*(\eta_{(i,j)}, \mu_{(i,j)}, \sigma^2_{(i,j)}, \boldsymbol{\eta}_{(-h)}|y)}{p^*(\eta_h, \mu_h, \sigma^2_h, \boldsymbol{\eta}_{(-h)}|y)} \frac{q(2\to 1)}{q(1\to 2)} \frac{1}{\prod_{i=1}^6 \pi(u_i)} \frac{1}{|J|}\right\}.$$

2.6.3 Updating $\boldsymbol{\gamma}$

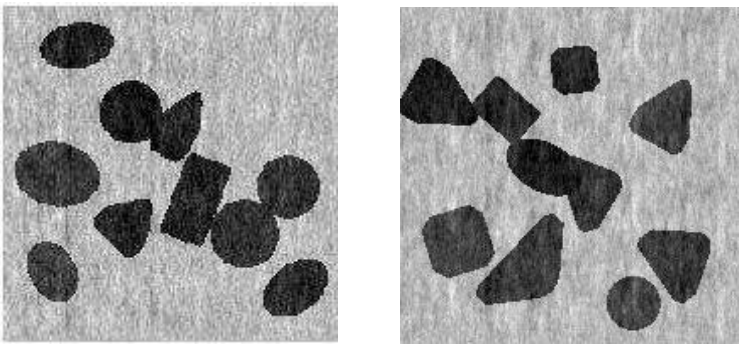
Random walk log-Normal proposal, $q(\gamma_i, \gamma_i^k) = \log - N(\log(\gamma_i^k), \Delta)$, is used to sample from the pseudo posterior distribution of $\boldsymbol{\gamma}$,

$$p^*(\boldsymbol{\gamma}|\boldsymbol{\Theta}, y) \propto \pi^*(\boldsymbol{c}, m|\boldsymbol{\gamma}, \mathbf{g^r}, \mathbf{s}, \boldsymbol{\theta}, \mathbf{T}) * \pi(\boldsymbol{\gamma}) = \exp\left\{-\gamma_1 m - \gamma_2 S(\boldsymbol{\eta})\right\} \pi(\boldsymbol{\gamma}).$$

2.7 Simulations

In this section, we use a simulation study to evaluate the performance of our proposed MCMC method. Two 200 by 200 pixels images ten number of objects each are generated from the prior distributions described in Section (4.1.1) with area interaction parameter $\gamma_2 = 40$ and $\gamma_2 = 10$ respectively. The pixels inside each object have constant mean, which is different from object to object. The covariance matrix is chosen from a CAR model with parameters very close to the extreme dependence. Two images with ten different sizes, rotations and center objects which belong to five different shape families, described in Section (2), are shown in Fig. 3.

At first, we compare the MCMC results from the proposed model with the results of the model that does not penalize the overlapping. More specifically, in the first case we considers γ_2 a random variable while in the second γ_2 is considered known and equal to zero. In both cases the parameter γ_1 is chosen constant and equal to 10. The

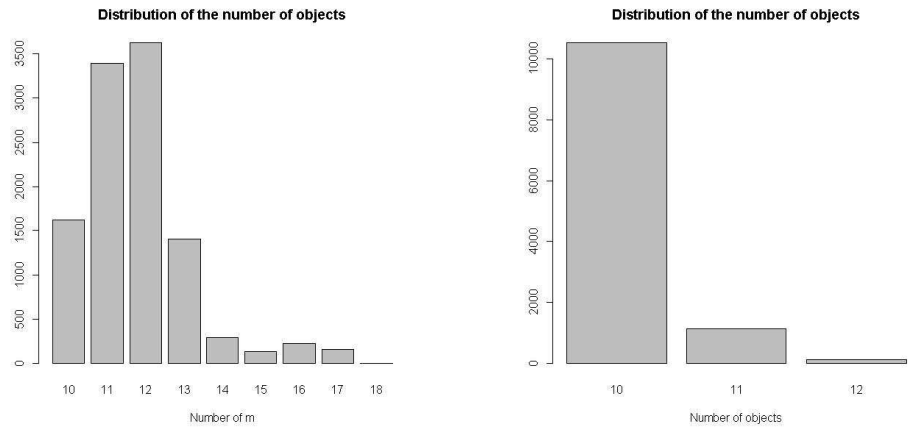


(a) $\gamma_2 = 40$ (b) $\gamma_2 = 10$

Figure 3: Simulation of two different images with $m = 10$ and their coresponding values for the interaction parameter γ_2

MCMC posterior distribution of m for image in Fig. 3(a), in a total of 12000 iterations, is recorded and presented for these two different cases in Fig. 4. The distribution of the number of objects m in the case of $\gamma_2 = 0$ is mostly a misspecification of the real image. In this case we have a sample of up to 18 objects, which almost doubles the original number of objects. An obvious overestimation of the number of objects in the posterior distribution occurs when we do not penalize the overlapping. On the other hand when we choose γ_2 as a random variable 90% of the posterior simulated number of objects represent the true number of objects. Treating γ_2 as unknown, in comparison with $\gamma_2 = 0$ yields a better fit and improves classification. For the case where γ_2 is fixed at a value different from zero the answer depends on how close the original and the assumed value of γ_2 are. If we fix the value of γ_2 in the range determined from the MCMC updates the results on the number of particles and shape analysis are not very different from the original values. Nevertheless, values outside the range can change the results dramatically. The same observations are true for the second simulated image.

After demonstrating the significance of the penalized overlapping, we move to the simulated posterior distribution of γ_2. For the two simulated images, the MCMC

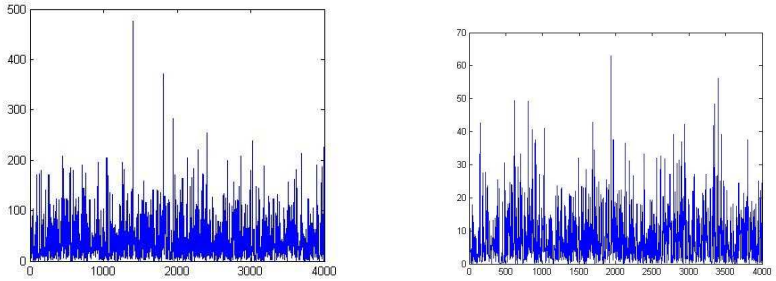


(a) $\gamma_2 = 0$ (b) γ_2 random

Figure 4: Distribution of the number of objects, m, for two different values of γ_2: (a) $\gamma_2 = 0$ and (b) γ_2 is considered random

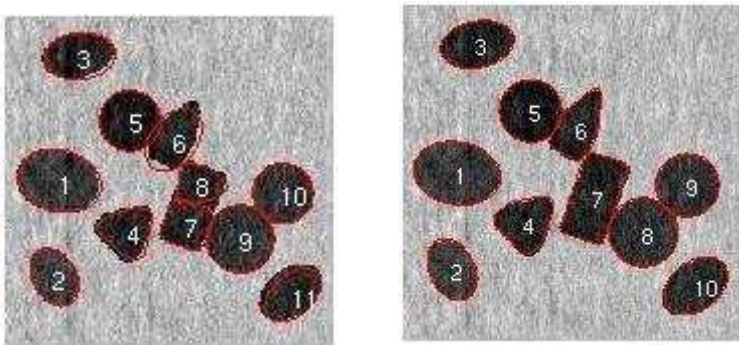
sampler for the posterior distribution of γ_2 are given in Fig. 5. From these simulations, we can see that the Markov chain mixes well and the posterior mean is close to the values we simulate the data. Values close to 40 are drawn for the first simulated image {Fig.-3(a)} while values close to 10 are drawn for the second simulated image. A general observation in the simulations is that the variance of the posterior distribution of γ_2 depends on the value of γ_2. For large values of γ_2 we observe relatively large posterior variance than it is for small values. Another significant observation is that there is a dependence on the accuracy and the variance of the posterior distribution of γ_2 on the number and size of objects. To investigate this phenomenon, we fixed the value of γ_2 but simulate images with different number of objects and sizes. As we increase the number and the size of objects, the posterior distribution of γ will be closer to the true value.

Finally, different moves are displayed by showing the results of two MCMC interactions. We can see the merge and split step in action in Fig. 6 and Fig. 7, respectively. In the absence of these move steps it would have required a large number of iteration



(a) γ_2 posterior (b) γ_2 posterior

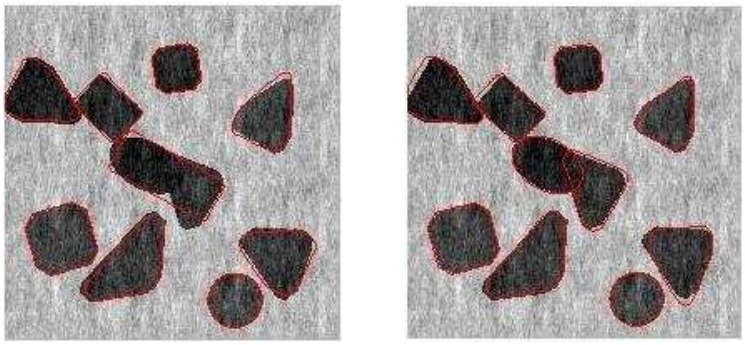
Figure 5: The last 4000 simulated values of γ_2 for the two different cases: (a) $\gamma_2 = 40$ and (b) $\gamma_2 = 10$



(a) 1000 iteration (b) 1500 iteration

Figure 6: Simulation of objects at (a) 1000 iteration (b) 1500 iteration. Except from the different movements a merge and a change template move has occurred

terms to arrive at this letter configuration. We present the two different move steps that occurred in the two simulated images. The 1000 and the 1500 MCMC iteration is given for the first image. In additional to different moves there is an obvious merge move step of 7^{th} and 8^{th} objects in Fig. 6(a) to 7^{th} object in Fig. 6(b). Also we display a split move step in the second simulated image. The 1400 and the 1600 MCMC iteration for the second image are given in Fig. 7(a) and Fig. 7(b). Not only an obvious split step has occurred but also we can see the different deviations of the boundaries which are related to the object representation parameters.



(a) 1400 iteration (b) 1700 iteration

Figure 7: Simulation of objects at (a) 1400 iteration (b) 1700 iteration. Except from the different movements a split template move has ocured

2.8 Application to gold nano particle analysis

Using the MCMC samples, we can obtain the distribution of the particle size, which is characterized by the area of the nanoparticle and the distribution of the particle shape. The aspect ratio, defined as the length of the perimeter of a boundary divided by the area of the same boundary, can be derived from the combination of size, shape and the pure parameters. The statistics of size, shape and aspect ratio are widely adopted in nano science and engineering to characterize the morphology of nanoparticles, and are believed to strongly affect the physical or chemical properties of the nanoparticles (El-Sayed, 2001; Nyiro-Kosa et al., 2009). For example, the aspect ratio is considered as an important parameter relevant to certain macro-level material properties because physical and chemical reactions are believed to frequently occur on the surface of molecules so that as the aspect ratio of a nanoparticle gets larger, those reactions are more active.

We apply our method to three different TEM images. The parameters that maximizes the posterior distribution (MAP) obtained from the (MCMC) are presented in detail. Our classification results of particular type are verified by our collaborators

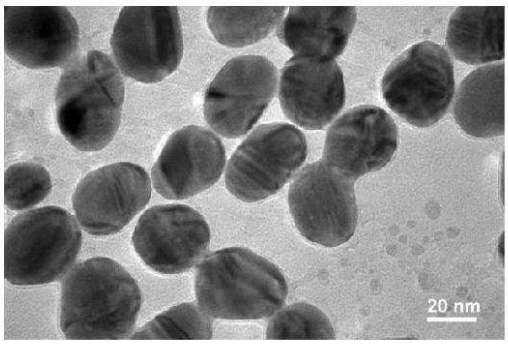
Table 1: MAP estimates of the parameters for the first six objects in Ex1

Object	Shape (T)	Center (x,y)	Size (s)	Rotation (θ)	g^r	Mean (μ)
1	E	(39.68, 32.72)	51.49	-0.21	1.14	50.64
2	E	(105.92, 105.92)	49.41	1.41	1.22	74.67
3	E	(175.79, 41.29)	47.20	1.36	1.12	62.55
4	E	(25.87, 221.72)	28.86	0.61	1.15	71.58
5	E	(39.89, 297.00)	49.98	0.83	1.13	64.58
6	C	(116.07, 362.30)	51.82	NA	NA	73.76

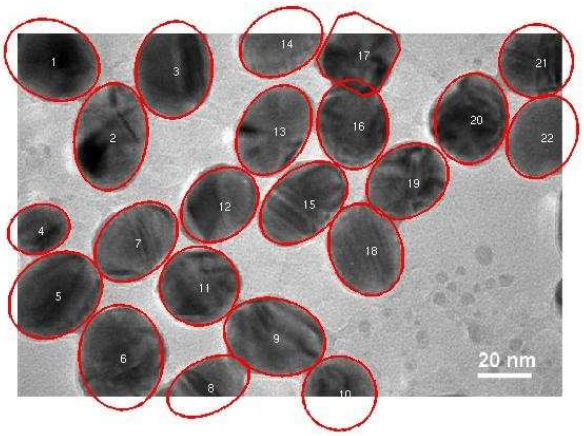
with domain expertise, this manual verification appears the only valid way for the time being. More than 95% of the nanoparticles in those images are classified correctly. This also includes the particles in the boundary as well as having overlapping regions. For completely observed objects, there is almost 100% correct classification.

We start our application with the image in Fig. 8.(a). Morphological image processing steps can be used to get an approximate count of the number of nanoparticles in the model. They also can be used in initializing the MCMC chains and in constructing proposal distributions required by the MCMC sampler. The morphological image processing we used in this dissertation has the following steps: (1) image filtering and segmentation, (2) determining the number of objects, (3) estimating location, size and rotation parameters. Because this morphological processing is not the subject of the present work, it is not presented in more details. After the initial values are obtained from the preprocessing step, all the five templates are randomly assigned for starting template specifications. The parameters drawn from the MCMC output that maximize the posterior namely, shape T, size s, rotation θ, random pure parameter g^r and mean intensity μ, are presented for the first six objects in Table 1.

In Fig. 8, we show the TEM image and MAP estimates of the parameters for 20,000 MCMC sample. In Fig. A-1, we present the parameters of s, g^r and μ that correspond to the MAP estimate for all the number of objects, m, corresponding



(a) TEM image Example-1 (Ex1)



(b) Object shapes at the maximum posterior of 20000 MCMC sample

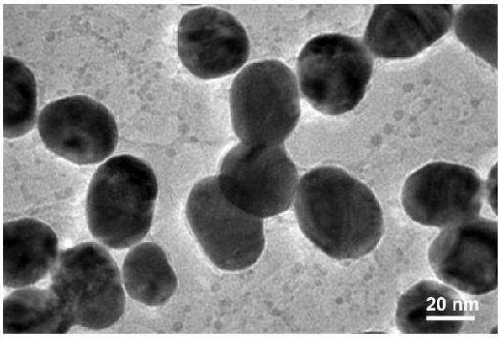
Figure 8: Object shapes sampled using MCMC in Ex1

Table 2: MAP estimates of the parameters for the first six objects in Ex2

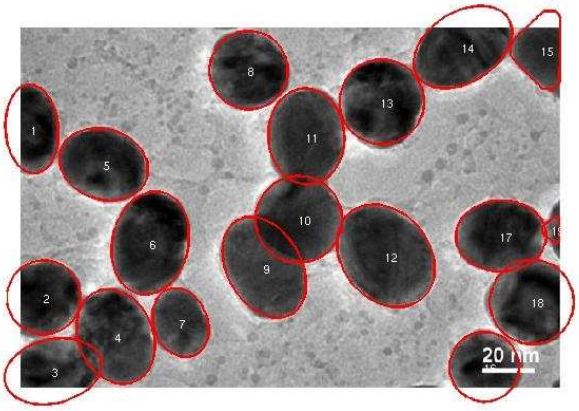
Object	Shape (T)	Center (x,y)	Size (s)	Rotation (θ)	g^r	Mean (μ)
1	E	(13.97, 256.78)	37.48	-1.51	1.2960	39.185
2	C	(27.44, 275.96)	41.04	NA	NA	42.969
3	E	(37.56, 314.44)	38.02	-0.29	1.2175	52.569
4	E	(106.40, 321.61)	47.44	-1.17	1.1591	60.605
5	E	(93.20, 413.87)	44.33	-0.36	1.1612	51.080
6	E	(146.67, 406.42)	49.63	-1.76	1.1621	44.617

to that value. Summary statistics of the shape parameters are given in Table 1. From the table and the histogram it is clear that the mean intensity is different from nanoparticle to nanoparticle, justifying our assumption of different means in Eqn. (3). We also obtain the posterior probability of the classification for each of the objects. This probability depends on the complexity of the shape of the object. For example, object 2 has been classified as an ellipse with probability 0.98 where as object 20 has been classified as an ellipse with probability .68 (circle with probability 0.32). In Table 1 (and in all the following tables of this chapter), we presented the classification with the highest posterior probability.

Our second application deals with a more complex image shown in Fig. 9(a). In this image at least 6 overlapping areas and at least 6 nanoparticles laying in the boundary are observed. More specifically: nanoparticles 1,2,3,14,15,16,18, and 19 lay in the boundary of the image while pairs $2-4$,$3-4$,$9-10$,$10-11$,$17-18$, and $10-12$ overlap. In this example, the overlapping is more complex and existing methods fail to represent the real situation. MAP estimates values for all the parameters are obtain after 20,000 MCMC iterations. Complex shapes have been classified accurately, see Fig. 9(b). For example, nanoparticle 18 has an incomplete image and it has been classified as a circle with posterior probability 0.77. The MAP estimates of the parameters drawn from MCMC namely, shape T, size s, rotation θ, random pure



(a) TEM image Example-2 (Ex2)



(b) Object shapes at the maximum posterior of 20000 MCMC sample

Figure 9: Object shapes sampled using MCMC in Ex2

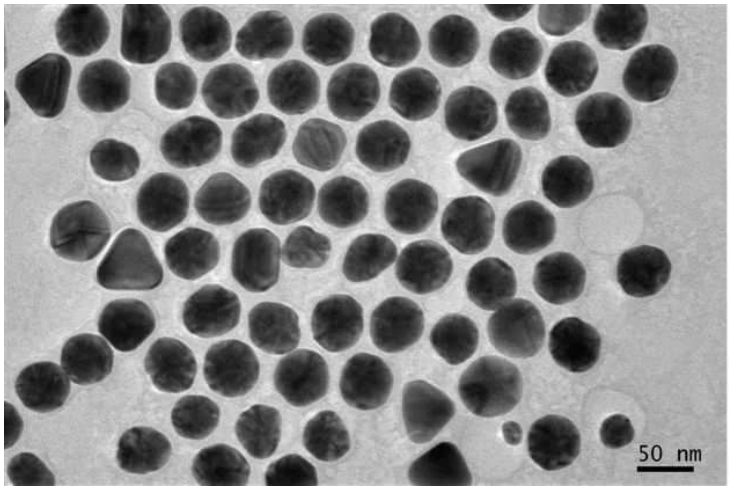
Table 3: MAP estimates of the parameters for the first six objects in Ex3

Object	Shape (T)	Center (x,y)	Size (s)	Rotation (θ)	g^r	Mean (μ)
1	E	(-3.11, 68.18)	12.43	-1.57	1.29	66.27
4	T	(35.53, 110.92)	25.82	1.38	2.32	49.33
12	T	(306.90, 225.73)	28.73	0.35	2.31	79.59
28	E	(219.91, 221.35)	24.09	1.53	1.14	68.19
51	T	(365.75, 352.49)	24.61	-1.46	2.25	63.29
57	T	(422.15, 139.28)	25.25	0.25	2.01	70.49

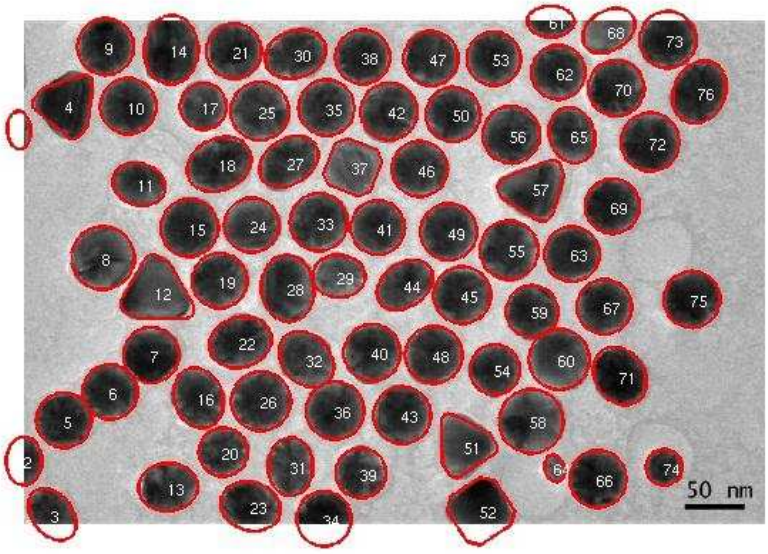
parameter g^r and mean intensity μ are presented for the first six objects in Table 2. In this application, 11 out of the 17 objects are ellipses (E) and 6 are circles (C) and one is triangle (TR). We also present the histogram of the MAP estimates parameters s, g^r and μ in Fig. A-2. Summary statistics of various shape parameters are given in Table 2. We see from the table that, our proposed algorithm captures triangles, circles etc. quite accurately.

Our next application deals with an image with 76 nanoparticles with 4 shapes, see Fig. 10(a). In this image, few objects have overlapping areas and at least 10 nanoparticles are laying in the boundary. Some objects do not have very clear shape like objects 29 and 31.

Different shapes are captured with different templates with the proposed method. In addition to the circles and ellipses which were successfully captured in the previous images, the triangles and squares are also captured accurately. Nanoparticles, 29 and 31 those have vague shapes are classified correctly, see Fig. 10(b). In this example, out of 76 nanoparticles, 47 are classified as a circle, 23 as an ellipse, 4 as a triangle and 2 as a square. Distribution of the various parameters of the identified objects are shown in Fig. A-3. In Table 3, we present all the triangular shapes in order to compare the pure parameter h_1. As we can see from the table, triangular shape nanoparticles 4 and 12 are closer to the equilateral triangle, with value close to $h_1 = 2.33$, while



(a) TEM image Example-3 (Ex3)



(b) Object shapes at the maximum posterior of 20000 MCMC sample

Figure 10: Object shapes sampled using MCMC in Ex3

triangular shape nanoparticle 51 and 57 have wider sides, since their $h_1 < 2.3$.

As a part of the verification process, we compare the accuracy of our method with that of the current practice used in nanoscience. In brief, the current practice is largely a manual process with support of image processing tools such as ImageJ Particle Analyzer (http://rsbweb.nih.gov/ij) and AxioVision (http://www.zeiss.com/), which have been popularly used for biomedical image processing. The results are shown in Figs. 11 and 12.

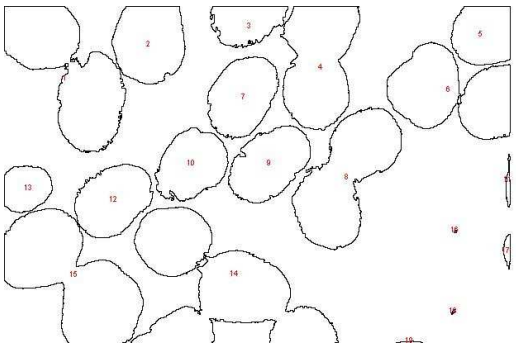


Figure 11: Objects identified by ImageJ in Ex1. Out of the 22 particles, 4 are recognized. Recognition rate = 18.18%

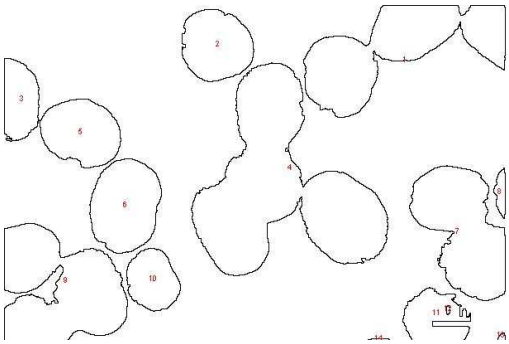


Figure 12: Objects identified by ImageJ in Ex2. Out of the 19 particles, 6 are recognized. Recognition rate = 35.58%

The manual counting process, subject to the application of the above imaging tools, is necessitated by the low accuracy of the autonomous procedures. For three

TEM images with overlaps among particles, our procedure recognized 95% of the total articles compared to the $20 - 50\%$ recognition rate of the ImageJ. Considering frequent occurrence of overlaps in the TEM images of nanoparticles, the existing software cannot be used as more than a supporting tool.

2.9 Concluding remarks

We adopted a Bayesian approach to image classification and segmentation simultaneously and applied it in TEM images of gold nanoparticles. We used marked point process to represent the nanoparticles in the image, where points represent the location of nanoparticles and marks represent their geometrical features. More specifically, we treated the nanoparticles in the image as objects, wherein the geometrical properties of the object were largely determined by templates and the interaction between the objects was modeled using the area interaction process prior. By varying the template parameters and applying operators such as scaling, shifting and rotation to the template, we modeled different shapes very realistically. In our current applications, we chose circle, triangle, square and ellipse as our templates. Other templates can be also constructed in the same framework. To solve the intractability of the posterior distribution we proposed a complex Markov Chain Monte Carlo (MCMC) algorithm which involves Reversible Jump, Metropolis-Hasting, Gibbs sampling and a Monte Carlo Metropolis-Hastings (MCMH) for the intractable normalizing constants in the prior. The first steps deal with simulating from a pseudo posterior distribution without involving the random normalizing constant. A generalized Metropolis-within-Gibbs sampling with a reversible jump step is used to simulate from a pseudo posterior distribution given the number of objects. Additionally, a reversible jump MCMC with the use of birth-death and merge-split moves is invoked on moving from state with different number of objects. Finally, we simulate from the intractable normalizing

constant posterior using Monte Carlo Metropolis-Hastings where the acceptance ration of the sample taken from the pseudo posterior is estimated by simulating from an auxiliary variable. We reported the posterior summary statistics of the shapes and the number of objects in the image. We successfully applied this algorithm to real TEM images with nanoparticles outperforming convention tools aided by manual screening.

CHAPTER III

REDUCED-DIMENSION HIERARCHICAL STATISTICAL MODELS

In the recent years a number of non-stationary covariance functions have been proposed to model the non-stationary Gaussian spatial processes. This ability to model with accuracy the variability, or heterogeneity, in the unknown process is of particular importance in environmental, geophysical, and other spatial datasets, in which domain knowledge suggests that the covariance structure may be non-stationary. However, the different proposed models usually require an expensive computational cost when we have to deal with large dimensional data.

Spatial deformations have been used to model non-stationary spatial processes in Sampson and Guttorp (1992); Schmidt and O'Hagan (2003); Anderes and Stein (2008) among others. Despite, the simplicity of the approach by maintaining the stationary structure of the covariance function the research on this approach has focused on multiple noisy replicates of the spatial function rather than the setting of one set of observations on which we focus here. Jun and Stein (2008) used a parametric non-stationary covariance function on the global scale which can reduce the computations when dealing with high dimensional gridded data. The method is particularly developed to deal with TOMS data of level 3 and 2, but computations can be facilitated only for level 3 data. Other approaches use different fixed set of basis functions that can be modeled on space e.g. wavelet, kernel etc.. Nychka and Royle (2002) applied a wavelet approach to produce non-stationary covariance function. In addition, Cressie and Johannesson (2008) modeled non-stationary covariance structure through a fixed rank approximation. Kernel convolution have been applied in several papers due to their simplicity to create non-stationary covariance functions and to

deal with the high dimensionality (Higdon (1998); Higdon et al. (2011); Calder (2008) among others). Fuentes (2002) introduced a different kernel based model, which is an extension of the finite decomposition method of Fuentes (2001). The spatial process is represented as a convolution of stationary processes. Paciorek and Schervish (2006) extended the non-stationary covariance function of the Kernel approach by providing a more general class of closed-form non-stationary covariance functions which are built upon familiar stationary covariance functions, e.g. Matérn. This non-stationary covariance function does not follow the usual kernel convolution method where we reduce also the computational cost. In this method we need to make the computations with the full covariance matrix.

In the recent literature there are various approximation techniques of the covariance functions that have been introduced in order to reduce the computational cost. The fixed rank approximation covariance matrix, Cressie and Johannesson (2008), is one of the most popular techniques which model also the non-stationary of the data. A fundamental limitation of this method is that implementation requires either independent replication of the spatial process or a parametric representation of the covariance function of the data in order to obtain a good estimation of the covariance matrix. Banerjee et al. (2008) proposed the predictive process which is as well a very popular technique. The predictive process is a reduction method based on hierarchical predictive Gaussian Process which captures the large scale spatial dependence. Their paper includes a simulation study to show that their reduction method can be used also in data with non-stationary covariance function define by Paciorek and Schervish (2006). Despite the success of the simulation study the method is not used to real data and the non-stationary regions are predetermined and known. A new method proposed by Sang and Huang (2011) simultaneously captures both the large and small scale spatial dependence. A reduced rank covariance technique is

proposed for the large scale covariance and tapering (or block) covariance is used to model the small scale. The method has been proven to work well in the case where a stationary known covariance function is assumed but its behavior has not been tried in high-dimensional data with non-stationary covariance function. Another reduction method, which produce a non-stationary covariance function, is the Bayesian Treed Gaussian Process Models, proposed by Kim et al. (2005) and Gramacy and Lee (2008). This method can model data with piecewise space-varying mean and non-stationary covariance function. However, the model is performing poorly for prediction in the boundary of the subregions. This is because the subregions considered cannot incorporate information from the entire space or neighboring subspaces. This is apparent also from the simulation study in section (3.5). Moreover, even if the data support this model, we may have computational problems to find the subregions and to compute the posterior distribution of the parameters inside each subregion. A non-stationary covariance function with a reduced computational cost should be considered.

In geophysical and environmental applications, it is common to have huge dimensional data with non-stationary covariance function. Total Ozone Matrix Spectrometer (TOMS) data are such an example. A detail description of the non-stationary of the data is given by Jun and Stein (2008), where it is clear a strong dependence of the covariance structure on latitude but not much dependence on longitude. The model proposed by Jun and Stein (2008) deals preliminary with Level 3 data since the computational cost of the method is very big to deal with Level 2 data.

We propose a model which separates the space with non-stationary covariance function into subregions with stationary covariance functions. The method is a combination of the use of the reduced covariance function proposed by Sang and Huang (2011) and a use of the non-stationary covariance function proposed by Paciorek and

Schervish (2006). The non-stationary covariance function proposed by Paciorek and Schervish (2006) links different reduced stationary covariance functions from separate subregions into a unique covariance function. The choice of this non-stationary covariance structure has the advantage of computational efficiency, since we can do independently the computations in each subregion. This model is applied to TOMS level 2 data where as we explain the non-stationary is coming though the latitude and other methods are difficult to be applied. In this chapter we chose predetermine subregions which are equal in length. We try to take as many as possible subregions. As it is demonstrated in the first simulation study when the covariance function is stationary or close to stationary the use of the non-stationary covariance function will not affect significantly the results. Moreover, when we have small differences of the covariance parameters inside a subregion the results will not change significantly if we consider a constant parameter inside this subregion.

3.1 Gaussian process models for spatial data sets

In this section, we present a summary of Gaussian process models for spatial data sets. Our presentation of Gaussian process models is based on the standard treatment in Banerjee et al. (2004) and Schabenberger and Gotway (2005).

3.1.1 Gaussian process

The basic geostatistical Gaussian model is of the form:

$$Z(s) = \mu(s) + w(s) + \epsilon(s), \tag{3.1}$$

where the process is decomposed in a mean part and two independent error processes, $w(s)$ and $\epsilon(s)$: $\epsilon(s)$ models the measurement error, also known as the nugget effect, which is usually modeled with an independent Gaussian process, $w(s)$, is introduced

to capture the spatial association and it is assumed to be a mean zero Gaussian spatial process. In a parametric approach the main issue is to choose a valid and appropriate correlation function, typically considering families of stationary processes.

The most common specification for $w(s)$ is $w(s) \sim GP(0, C(\cdot\,,\cdot\,))$, a zero-mean Gaussian process with a valid stationary covariance function $C(s', s)$. A widely used, flexible choice of correlation function is the isotropic Matérn family of correlation functions (see, e.g., Stein (1999)), $C(s', s) = \sigma^2\rho_0(\mathbf{h}; \nu, \phi)$, where $\mathbf{h} = s - s'$ is the vector of the difference between the coordinates, σ quantifies the spatial variance, ϕ quantifies the correlation range and ν the smoothness of the process path.

An extension of the isotropic stationary correlation function is to include anisotropic correlation function where spatial association depends upon the separation vector between locations. A special case which also is the most prominent among the different anisotropies models is the *geometric anisotropy*. This refers to the case where the coordinate space can be linearly transformed to an isotropic space. In general for $\Re^d$ space,

$$\rho(\mathbf{h}; \phi) = \rho_0(||L\mathbf{h}||; \phi)$$

where L is the $d \times d$ matrix of the linear transformation. Another way to see the geometric anisotropy is by substituting the existing Euclidian distance $||\mathbf{h}||$ with a Mahalanobis distance $\mathbf{h}'B^{-1}\mathbf{h}$ in the isotropic covariance function, where $B = L'L$. Because we can have the same value of the matrix $B * \phi$ with different values of B and ϕ we ignore ϕ and when we refer to B we refer to $B * \phi$, for identification reasons. For a diagonal marix B with equal values on the diagonals the correlation function reduces to the isotropic correlation function.

In this chapter we prefere to decompose B into a diagonal eigenvalue matrix Λ

and eigenvector matrix $\Psi(\theta)$ as: $B = \Psi(\theta)\Lambda\Psi(\theta)$. This will facilitate to distinguish the dependence in space and the computations.

3.1.2 Modeling the mean $\boldsymbol{\mu}$

Since we focus on modeling the covariance structure of this data, we should somehow filter the data and make the process close to mean zero. Spherical harmonics provide a natural basis for capturing the large-scale patterns in the glob (Jun and Stein 2008). Specifically, we regress the ozone levels with $X_m^n(\sin\vartheta, \phi)|n = 0, 1, 2, \ldots, m = -n, \ldots, n$ for $n = 12$. This will make the Eqn. (3.1):

$$Z = X\beta + W + \epsilon, \tag{3.2}$$

where X is the spherical harmonics basis matrix, W is the vector of spatial error and ϵ is the vector of random error.

In what follows we give a brief review on the spherical harmonics basis functions.

3.1.2.1 Spherical harmonics

Laplace's equation is a linear second-order differential equation. This common and important equation can describe many problems of theoretical physics, e.g. electromagnetic phenomena, hydrodynamics, heat flow and gravitation (Arfken (1970)), and is expressed in its most general form with the squared del operator which is applied on a function f and in the $\Re^3$ Cartesian Coordinates take the form:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = 0,$$

and in three dimentional spherical Coordiantes (r, ϑ and φ):

$$\nabla^2 f = \frac{1}{r^2}\frac{\partial}{\partial r}\big(\frac{\partial f}{\partial r}\big) + \frac{1}{r^2\sin(\vartheta)}\frac{\partial}{\partial\vartheta}\big(\sin(\vartheta)\frac{\partial f}{\partial\theta}\big) + \frac{1}{r^2\sin^2(\vartheta)}\frac{\partial^2 f}{\partial\varphi^2} = 0. \tag{3.3}$$

The spherical harmonics are then an orthogonal set of solutions to Laplace's equation of three dimensions in spherical coordinates. If only the angular portion is considered the resulting functions are called surface spherical harmonics. Since only surface spherical harmonics are of interest for the presented work the term 'spherical harmonics' refers hereafter to surface spherical harmonics.

The most common way to get particular solutions of partial differential equations is by the method of separation of variables. The method of separation of variables is applied twice and the solution is a product of trigonometric functions with the associated Legendre functions:

$$X_m^n = Ne^{in\varphi}P_m^n(\cos(\theta))$$

where X_m^n is called a spherical harmonic function of degree m and order n. N is a normalization constant, $e^{in\varphi}$ describes the trigonometric functions of the product (function of the longitude angle ϕ), and $P_m^n(\cos(\theta))$ the associated Legendre function.

3.1.3 Modeling the covariance matrix

Since the global TOMS data are non-stationary, we cannot use a stationary Matérn covariance function. A detail description of the non-stationary nature of the data is given by Jun and Stein (2008), where it is clear a strong dependence of the covariance structure on latitude but not much dependence on longitude. We can also see this dependence in the results of this chapter. A novel parametric non-stationary covariance matrix is proposed by Paciorek and Schervish (2006). The non-stationary covariance function proposed is an extension of the stationary parametric case built upon familiar stationary covariance functions with geometric anisotropy. Specifically, if an isotropic correlation function, $\rho_0()$, is positive definite on $\Re^d$ for

every $d = 1, 2, \ldots$, then the function, $\rho_{NS}(||\mathbf{h}||)$, defined by

$$\rho_{NS}(s_i, s_j) = |B_i|^{\frac{1}{4}}|B_j|^{\frac{1}{4}}\Big|\frac{B_i + B_j}{2}\Big|^{\frac{1}{2}}\rho_0(Q_{ij}), \tag{3.4}$$

with $Q_{ij} = \sqrt{(s_i - s_j)'\big((B_i + B_j)/2\big)^{-1}(s_i - s_j)}$ used in place of $||\mathbf{h}||$, is a valid non-stationary correlation function and it is positive definite on $d = 1, 2, \ldots$. The proof of the validity is a simple application of Schoenberg theorem (1938) (Paciorek and Schervish (2006)).

The result applies to any positive definite correlation function in the Euclidean space of a particular dimension. This means that we can use any known stationary correlation function as the power exponential, rational quadratic, and Matérn to construct a unique and valid non-stationary covariance function.

In the referred paper the case where not only the smoothness parameters are different for different region but also the variances is not mentioned. This can be done easily by just considering $C(s', s) = \sigma_s\sigma_{s'}\rho_{NS}(s, s'; \nu, \phi)$. Where σ_s is the function of the standard deviations in different regions and ρ_{NS} is the covariance proposed correlation function. More specifically a non-stationary version of the Matérn correlation function will have the form:

$$\rho_{NS}(s_i, s_j) = \sigma_{s_i}\sigma_{s_j}\frac{1}{\Gamma(\nu)2^{\nu-1}}|B_i|^{\frac{1}{4}}|B_j|^{\frac{1}{4}}\Big|\frac{B_i + B_j}{2}\Big|^{\frac{1}{2}}\big(2*\sqrt{\nu Q_{ij}}\big)^{\nu}K_{\nu}\big(2*\sqrt{\nu Q_{ij}}\big). \tag{3.5}$$

This non-stationary covariance model can also cover the case where the parameters change over space for every observation. The authors have developed an MCMC algorithm for this particular case but this approach is expensive in time and it is hard to be considered in a computation reduction paper. Instead the assumption of stationary over subspaces will facilitate the computation. This is the approach which we consider in this chapter.

Let's suppose that we partition the input space into D non-overlapping regions:

$r_{\xi}{}_{\xi=1}^{D}$. Each region r_ξ contains covariate and data $\{X_\xi, Z_\xi\}$, consisting of n_ξ observations and the generative GP model for every of the regions is: For the case where we have D different region we can write the covariance matrix as:

$$\Sigma = C + D_{\tau^2} = \begin{pmatrix} \sigma_1^2 R_{11} & \sigma_1\sigma_2 R_{12} & \dots & \sigma_1\sigma_D R_{1D} \\ \sigma_2\sigma_1 R_{21} & \sigma_2^2 R_{22} & \dots & \sigma_2\sigma_D R_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_D\sigma_1 R_{D1} & \sigma_D\sigma_2 R_{D2} & \dots & \sigma_D^2 R_{DD} \end{pmatrix} + \begin{pmatrix} \tau_1^2 \boldsymbol{I} & 0 & \dots & 0 \\ 0 & \tau_2^2 \boldsymbol{I} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \tau_D^2 \boldsymbol{I} \end{pmatrix},$$

where Σ_E is a diagonal variance matrix with equal variance for separate. The likelihood will have the form:

$$Z|\Theta \sim N(X\beta, \Sigma). \tag{3.6}$$

It is obvious that if we know the partitions we can find exactly the covariance function.

One possible approach to partition the space is the Treed partitioning. Treed partition models typically divide up the input space by making binary splits on the value of a single variable. Since variables may be revisited, there is no loss of generality by using binary splits, as multiple splits on the same variable will be equivalent to a non-binary split. The model is very similar to the Gaussian Tree process (Denison et al. (1998); Chipman et al. (1998, 2002); Gramacy and Lee (2008)) with the difference that our tree model does not assume independent observations between the subregions of a tree. Moreover, only the parameters related to the covariance function are assumed to depend on the tree.

Another characteristic of the ozone data is that the non-stationarity of the covariance matrix is coming through different latitude (Jun and Stein (2008)). This will facilitate our search for the tree by dividing the space only for different latitude.

3.1.4 Modeling the anisotropy on earth

The Chordal distance (more explicitly the great circle chordal distance) is a very good spherical distance which can model the distance in earth in a natural way producing isotropic covariance matrices. However its form makes it difficult to extend to the anisotropic covariance matrices. On the other hand, if we ignore the sphericity of the earth and work only with the two dimension Euclidean distance in longitude and latitude we will measure distances which are not a good representation of the reality.

To overcome these difficulties we propose the tunnel distance or alternatively the distance in a three dimension using Cartesian coordinates. The simple tunnel distance is the Euclidian distance between two points in a three dimensional space. Earth belongs to a 3D space and this is a natural way of representing the distance. The tunnel distance may be calculated as follows for the corresponding sphere with radius r, by means of Cartesian subtraction:

$$\Delta X = r * (\cos(\phi_f)\cos(\lambda_f) - \cos(\phi_s)\cos(\lambda_s)); \tag{3.7}$$

$$\Delta Y = r * (\cos(\phi_f)\sin(\lambda_f) - \cos(\phi_s)\sin(\lambda_s)); \tag{3.8}$$

$$\Delta Z = r * (\sin(\phi_f) - \sin(\phi_s)). \tag{3.9}$$

$$D_h^I = \sqrt{(\Delta X)^2 + (\Delta Y)^2 + (\Delta Z)^2},$$

where $D^I(s, s')$ is the "isotropic" distance in $3D$ Cartesian space and is the Euclidian distance of the two points. If we want to generalize the distance and make it anisotropic we use the Mahalanobis distance in the Cartesian coordinates. By this extension we can construct geometrical anisotropy covariance functions, so the spatial

correlation between two observations will not depend only on the absolute distance but also upon the separation vector between their locations.

Let $\Delta = (\Delta X, \Delta Y, \Delta Z)$ be a vector of the differences of the coordinates of two points s and s'. The anisotropic distance preferred in this dissertation is:

$$D^{NI}(s, s') = \sqrt{\Delta' B^{-1} \Delta},$$

where B is a 3×3 positive definite symmetric matrix which will determine the range and the angles of the range parameters.

For the isotropic Matérn covariance matrix we can substitute ϕ with $B = diag(\phi)$. If we want to separate the longitude and latitude range parameter we choose

$$B = \begin{pmatrix} \phi_1 & 0 & 0 \\ 0 & \phi_1 & 0 \\ 0 & 0 & \phi_2 \end{pmatrix},$$

where ϕ_1 corresponds to the longitude range parameter and ϕ_2 corresponds to the latitude range parameter.

The nonstationary covariance matrix in 2.3 assumes that each location, s_i, has a Gaussian kernel with covariance (kernel) matrix, $B_i = B(s_i)$. For a better representation, interpretation and compuational eficiency we decompose B_i into $B_i = \Psi_i \Lambda_i \Psi_i'$ where Λ_i is the matrix of eignevalues, $\lambda_1(s_i), \lambda_2(s_i)$ and $\lambda_3(s_i)$, and Ψ_i is an eigenvector matrix (rotation matrix) constructed as described below.

The rotation matrix in 3D can be represented in various ways. We chose the three basic (gimba-like) rotation matrices which rotates vectors in the x, y, or z axis, in three dimension. We write $\Psi(s_i) = \Psi_x(\theta_1(s_i))\Psi_y(\theta_2(s_i))\Psi_z(\theta_3(s_i))$ where $\Psi_x(\theta_1(s_i))$, $\Psi_y(\theta_2(s_i))$, and $\Psi_z(\theta_3(s_i))$ represent the rotation matrix for each dimention. Each embedding leaves one direction fixed, which in the case of 3×3 matrix is the rotation axis. The three rotational matrixes analytically are:

$$\Psi_x(\theta_1(s_i)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_1(s_i)) & -\sin(\theta_1(s_i)) \\ 0 & \sin(\theta_1(s_i)) & \cos(\theta_1(s_i)) \end{pmatrix},$$

$$\Psi_y(\theta_2(s_i)) = \begin{pmatrix} \cos(\theta_2(s_i)) & 0 & \sin(\theta_2(s_i)) \\ 0 & 1 & 0 \\ -\sin(\theta_2(s_i)) & 0 & \cos(\theta_2(s_i)) \end{pmatrix},$$

$$\Psi_z(\theta_3(s_i)) = \begin{pmatrix} \cos(\theta_3(s_i)) & -\sin(\theta_3(s_i)) & 0 \\ \sin(\theta_3(s_i)) & \cos(\theta_3(s_i)) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Each of these basic vector rotations typically appears counter-clockwise when the axis about which they occur points toward the observer, and the coordinate system is right-handed. Ψ_z, for instance, would rotate toward the y-axis a vector aligned with the x-axis. This is similar to the rotation produced by the $2 - D$ rotation matrix.

3.2 Reduction models

In this section we review three existing methods of approximating the covariance functions that allow rapid computation of the likelihood-based parameter estimation and spatial prediction, namely, the reduced rank, the tapering and the full-scale covariance approximation.

3.2.1 The predictive process model: reduced rank approximation

Reduced rank methods approximate the spatial process $w(s)$ in (1) by a process $w_l(s)$ that lies in a fixed, finite-dimensional space. Since the resulting covariance matrix of the data has a fixed rank, great computational savings can be achieved for both likelihood inference and spatial prediction.

The reduced rank approximation can be motivated through the Karhunen-Lóeve expansion of the spatial process (K-L expansion; Baker (1977)). Suppose the domain D of the process $w(s)$ is a compact set. Under certain conditions on the covariance

function $C(s, s_0)$, the Karhunen-Lóeve expansion decomposes $w(s)$ into a countable orthogonal series $z(s)$ as:

$$w(s) = \sum_{i}^{\infty} \sqrt{\lambda_i}\phi_i(s)z(s),$$

where λ_i are the descending values of eigenvalues and $\phi_s(s)$ is the eigenfunction which corresponds to the λ_i eigenvalue. The eigenvalue-eigenfunction pairs are solutions to the integral equation,

$$\int C(s', s)\phi_i(s)p(s)ds = \phi_i(s')\lambda_i\phi_i(s'), \tag{3.10}$$

where $p(s)$ is the distribution of the locations s and it is usually assumed to be constant and eventually ignored. The eigenfunctions are assumed to be orthogonal so that $\int \phi_i(s)\phi_j(s)ds = \delta_{ij}$, where $\delta_i j$ is the Kronecker delta. To solve this equation, Williams and Seeger (2001) sample m knots from $p(s)$ and approximate the above second order Freedholm equation with the discrete solution given by these m knots. Because the distribution is uniform, a better representation is by fixing the knots to cover the entire region equivalently. Let's consider a set of knots $\boldsymbol{S}^* = s_1^*, \ldots, s_m^*$. The discrete form of the above equation is:

$$\frac{1}{m}\sum_{k}^{m} C(s', s_k^*)\phi_i(s_k^*) \approx \lambda_i\phi_i(s'). \tag{3.11}$$

The reduced rank predictive process method will approximate the real model by:

$$Z(s) = \mu(s) + w(s) + \epsilon(s) \approx \mu(s) + w_l(s) + \xi(s) + \epsilon(s) = \mu(s) + W_l(s) + \epsilon'(s). \tag{3.12}$$

where $W_l(s) = C(s, S^*)C(S^*, S^*)w^*$ and w^* denote the realization of $w(s)$ at the m knots in S^*. For more information regarding predictive process and Noÿstrom method see, Sang and Huang (2011), Banerjee et al. (2008) and Williams and Seeger (2001) among others.

From the above equation it is obvious that except the fact that we lose some information we will increase the variance of the nugget error and most probably decreasing the variance of the spatial error. This is the reason why this method will in general produce weak estimations for the spatial variance and overestimate the nugget variance. The reduced rank approximation is also inaccuracy in representing local/small scale dependence (Stein (2008); Finley et al. (2009)). This can also be seen from the fact that the knots considered to reduce the rank are usually further apart from the real observations.

Except from the predictive process reduce rank the recent literature is rich in reduced rank methods. For example the fixed rank kriging (FRK) proposed by Johannesson et al. (2007); Cressie and Johannesson (2008) is one of the most famous reduced rank methods. Usually these methods are not parametric and depend on an the estimation of the covariance matrix.

3.2.2 Sparse matrix approximation

Another approach is to approximate the data covariance matrix by a sparse matrix and then employ the sparse matrix algorithm to achieve computational efficiency. If one believes that distant pairs of observations are uncorrelated, then one can use a compactly supported covariance function to model the spatial dependence (Gneiting (2002)). A common technique to achieve sparseness in the covariance tapering function can be found in (Genton and Nychka (2006) and Kaufman et al. (2008)). Another technique to achieve a sparse covariance matrix is by constructing subregions which are independent from each other. Independent subregions methods are faster than tapering but they lack of accuracy.

Let $h = x - x*$ and $K_0(h;\theta)$ denote the original covariance function for a stationary random field. Consider a tapering function $K_{taper}(h;\gamma)$ which is an isotropic

correlation function. The tapered covariance function is defined as

$$K_1(h;\theta,\gamma) = K_0(h;\theta)K_{taper}(h;\gamma),\ h > 0.$$

According to the Schur product theorem (Horn and Johnoson (1985), section 7.5), the tapered covariance function is positive semi-definite and thus a valid covariance function.

3.2.3 The full-scale covariance approximation

A new approach proposed by Sang and Huang (2011) combines the ideas of the reduced-rank process approximation and the sparse covariance approximation. The new approximation take advantages of both approaches while overcomes their individual shortcomings. This new method is called "full-scale" approximation of the covariance because of its capability of providing high quality approximations at both the small and large spatial scales. In short, we first should decompose the spatial Gaussian process into two parts: a reduced rank process to characterize the large scale dependence and a residual process to capture the small scale spatial dependence that is unexplained by the reduced rank process. We then obtain sparse covariance approximation of the residual process using covariance tapering or block covariance matrix. Since the residual process mainly captures the small scale dependence and the tapering has little impact on such dependence other than introducing sparcity, the error of the new approximation is expected to be small.

For the spatial process $w(s)$ as in Eqn. (3.1), consider the decomposition:

$$w(s) = w_l(s) + w_r(s),$$

where $w_l(s)$ is a reduced rank approximation of $w(s)$ and $w_r(s) = w(s) - w_l(s)$ is the residual of the approximation.

For m knots at a fix set of locations S^* and process realization $\boldsymbol{w}^*$, the predictive process can be expressed as:

$$w_l(s) = C(s, S^*)C(S^*, S^*)^{-1}\boldsymbol{w}^*$$

with covariance function: $C_l(s, s') = C(s, S^*)C(S^*, S^*)^{-1}C(s', S^*)'$, for every s and s'. The exact residual of the approximation is $w_r(s) = w(s) - w_l(s) = w(s) - C(S^*, s)C(S^*, S^*)^{-1}\boldsymbol{w}*$ and its covariance function is:

$$C_r(s, s') = C(s, s') - C(s, S^*)C(S^*, S^*)C(s', S^*)'$$

A short scale approximation is applied to the residual covariance matrix C_r which leaves it sparse or block diagonal. In the case where we apply the tapering technique we have:

$$C_s(s, s') = (C(s, s') - C_l(s, s'))K_{tapering}(s, s'; \gamma),$$

which is a valid covariance function with compact support. By putting things together we will have an approximation of the covariance matrix C as:

$$C = C_l + C_r \approx C_l + C_s = C^a.$$

C^a is called the full-scale approximation covariance matrix and provides a valid covariance function, Sang and Huang (2011).

The approximation of the covariance matrix as it is described above will facilitate the computations of the likelihood or posterior by applying the well known Sherman-Woodbury-Morrison formula for inverse matrices.

3.3 Bayesian inference with predetermined subregions

The Bayesian inference for the model parameters begins with assigning prior (hyperprior) distributions to the model parameters (hyperparameters). In this section

we assume that the partitions are given but not independent.

3.3.1 Prior specification

For each partition we will follow the standard method for prior specifications (Banerjee et al. (2004)) and assume independence between parameters of different bands and vague but proper priors. For β we chose a Normal distribution prior with very large variance to make it close to non-informative. The number of the parameters in the covariance will depend on the number of subregions and the choice of the stationary covariance function within the subregions. For simplicity, we chose the same prior specification for the covariance parameters in different subregions. An Inverse Gamma prior for the model error σ_i^2 and the nugget error variance τ_i^2. The parameters of the prior in the variance components are chosen such that the Inverse Gamma distribution will have a big variance and a reasonable guess of mean. Prior specifications for the range parameters will depend upon the choice of correlation function and also the anisotropic nature of the spatial dependence.

In the isotropic case we have only one range parameter, ϕ_i, for each subregion and its prior specifications will depend upon the choice of correlation function. In the case of Exponential distribution we can use more specific priors for $\phi_i \sim IG(2, b)$ where $b_0 = \rho_0/(-2\ln(0.05))$ and $\rho_0 = \max_{i,j} |s_i - s_j|$ (see Banerjee et al. (2004) and Schmidt and O'Hagan (2003)), for more details). In other words $\pi(\phi_j) \propto \phi_j^{-3} e^{-b/\phi_i}$.

A second approach is to take a reference-type prior for the parameters of the covariance. In this case we follow the reference analysis proposed by Berger et al. (2001). They proposed and recommended the use of the reference prior for the parameters of the correlation function because the reference prior always yields a proper posterior, in contrast to other noninformative priors. This prior is computationally more challenging and it is not suggested in this dissertation.

For the anisotropic covariance function we can use two kinds of prior distribution: i) Wishart prior for the matrix B as it is given in Banerjee et al. (2004) and ii) prior specification for the three eignevalues, λ_1, λ_2 and λ_3 , and three rotation parameters, θ_1, θ_2 and θ_3. In the second case a noninformative uniform prior on $(0, \pi]$ is chosen for every θ_j while prior specification for λ_j are given by an Inverse Gamma distribution with big variance and mean close to the mean of ϕ from previous studies.

3.3.2 Posterior inference

To facilitate notation we will refer to all the spatial range parameters as $\boldsymbol{\phi}$. Let $\Omega = [\beta, \boldsymbol{\phi}, \boldsymbol{\tau}, \boldsymbol{\sigma}]$ denote collectively the model parameters. The MCMC method is used to draw samples of the model parameters from the posterior:

$$p(\Omega|Data) = p(Z|\Omega)p(\beta)\prod_{i=1} p(\phi_i)p(\tau_i)p(\sigma_i). \tag{3.13}$$

We use Gibbs sampling to sample from the joint distribution. Sampling proceeds by first updating β from an $MVN(\mu_{\beta|.}, \Sigma_{\beta|.})$ distribution with the covariance matrix:

$$\begin{aligned} \Sigma_{\beta|.} &= [\Sigma_{\beta_0} + X'\{\Sigma\}^{-1}X] \\ &\approx [\Sigma_{\beta_0} + X'\{C_l + C_s + D_{\tau^2}\}^{-1}X], \end{aligned} \tag{3.14}$$

and mean

$$\begin{aligned} \Sigma_{\beta|.} &= \Sigma_{\beta|.}[\Sigma_{\beta_0}\mu_{\beta_0} + X'\{\Sigma\}^{-1}Z] \\ &\approx [\Sigma_{\beta_0} + X'\{C_l + C_s + D_{\tau^2}\}^{-1}Z], \end{aligned} \tag{3.15}$$

where X is the matrix of the basis function for the whole region, Z are the total observations, μ_{β_0} and Σ_{β_0} are the mean and covariance matrix of the prior distribution of β, and C_l, C_s and τ_i^2 are defined above. For the parameters $\boldsymbol{\phi}, \boldsymbol{\sigma}, \boldsymbol{\tau}$ which do not have closed form posterior conditional distributions, we will need to draw samples using Metropolis-Hasting steps (Gelman et al. (2004)). Following the MCMC sampling, posterior inferences such as posterior means and credible intervals are then made by computing summaries of the posterior samples.

3.3.3 Spatial prediction

For spatial prediction, we consider two approaches. One is the classical kriging method, i.e., the spatial best linear unbiased prediction (BLUP) given the MCMC mean parameters, and the other is the Bayesian prediction. Following Sang and Huang (2011) the BLUP prediction in location s_0 is given by:

$$Y(s_0) = x^T(s_0) + h^T(s_0)(C_l + C_s + D_{\tau^2})^{-1}(\mathbf{Y} - \mathbf{X}\beta), \tag{3.16}$$

where $h^T(s_0) = [C_l(s_0, s_i) + C_s(s_0, s_i)]_{i=1}^N$ and the mean square prediction error is:

$$Y(s_0) = \sigma^2 - h^T(s_0)(C_l + C_s + D_{\tau^2})^{-1}h(s_0) + \tau^2. \tag{3.17}$$

In the Bayesian approach we take a similar approach but now we have to compute randomly the value from MCMC values. $Y_{(s_0)}^{(l)} \sim p[Y_{(s_0)}|\Omega^{(l)}, Y]$ where $\Omega^{(l)}$ is the l^{th} sample from the MCMC posterior values.

3.4 Implementation

Since the global data are huge in dimension and non-stationary, the number of knots used in the predictive process and the full-scale approximation should sufficiently cover the globe. It is obvious that we will need a lot of knots to cover the whole globe and also we need to store a $N \times N$ covariance matrix. Despite the fact that we reduce the computational cost by using the full rank approximation or the predictive process we still need to store a $N \times N$ covariance matrix. Every MCMC iteration requires an approximate inversion of the covariance matrix using a lot of knots and storing a very big covariance matrix, e.g. $2.5 * 10^5$. This is usually very expensive and computationally intensive. To overcome these difficulties, we assume that the partitions are independent of each other. Since the subregions consist of

relatively large data, the assumption of independence will not change the posterior distribution of the parameters in the model a lot. This observation is crucial in reducing the computational cost without really losing in the accuracy of computing the posterior distribution. For more details on the efficiency this approach see also the first simulation study.

3.4.1 MCMC for the parameters

When we sample from the posterior distribution we assume that the partitions are given and independent of each other. Let $\Omega = [\beta, \boldsymbol{\phi}, \boldsymbol{\tau}, \boldsymbol{\sigma}]$ denote as above collectively the model parameters. We use Gibbs sampling to sample from an approximate joint distribution.

Independent MCMC's are applied to draw samples for the parameters $[\phi_i, \tau_i, \sigma_i]$ for every subregion. More specifically we will draw samples using Metropolis-Hasting steps as above for every subregion. In each of these draws, we use full rank or predictive process approximation for the covariance matrix and the draws are considered to come from an approximate posterior distribution.

We already know the distribution of β which $MVN(\mu_{\beta|.}, \Sigma_{\beta|.})$. Since the computational cost is huge we use the independent subregion logic and apply the covariance approximation.

We draw β from a normal distribution with the covariance matrix:

$$\begin{aligned} \Sigma_{\beta|.} &= [\Sigma_{\beta_0} + X'\{\Sigma\}^{-1}X] \\ &\approx [\Sigma_{\beta_0} + \sum_{i}^{D} X_i'{\Sigma_i}^{-1}X_i] \\ &\approx [\Sigma_{\beta_0} + \sum_{i}^{D} X_i'\{C_{l,i} + C_{s,i} + \tau_i^2 I_i\}^{-1}X_i], \end{aligned} \tag{3.18}$$

and mean

$$\begin{aligned}\Sigma_{\beta|.} &= \Sigma_{\beta|.}[\Sigma_{\beta_0}\mu_{\beta_0} + X'\{\Sigma\}^{-1}Z] \\ &\approx [\Sigma_{\beta_0} + \sum_{i}^{D} X_i'{\Sigma_i}^{-1}Z_i] \\ &\approx [\Sigma_{\beta_0} + \sum_{i}^{D} X_i'\{C_{l,i} + C_{s,i} + \tau_i^2 I_i\}^{-1}Z_i],\end{aligned} \tag{3.19}$$

where X_i are the spherical basis functions which correspond to the i^{th} band, Z_i are the observations corresponding to that band, μ_{β_0} and Σ_{β_0} are the mean and covariance matrix of the prior distribution of β, and $C_{l,i}, C_{s,i}$ and τ_i^2 are defined above.

3.4.2 Prediction

In contrast with the estimation of the parameters, the prediction of the data it is sensitive to the assumption of independence. As it is clear from the first simulation study when we ignore information from other subregions the prediction will be poor in the boundary regions. Especially in the boundaries, we should incorporate information from other subregions. If we can compute with the reduction techniques the covariance matrix for all the data then we can make the prediction with the full covariance matrix. In practice we were unable to store a matrix of $200,000 \times 200,000$ every time, and for that reason we suggest other approaches to be able to perform the prediction as accurately as possible.

One possible solution to the computational problem is to take only data from the neighboring subregions. For example to predict values from the i^{th} subregion we use data from the $(i-1), i$ and $(i+1)$ subregions. This is a fast and effective way to predict with high accuracy even in the boundaries of the subregions.

3.5 Simulations

In this section, we conduct a simulation study to evaluate the performance of our proposed MCMC method. For a better representation we generate values from

two Gaussian processes with non-stationary Matérn covariance function with two different subregions as it is described in section 2.3. The extension to the case with more subregions is obvious.

In the first simulation study we test the efficiency of using the non-stationary covariance function in comparison with the stationary or with the use of separate and independent subregions covariance functions. In the second simulation study we test the efficiency of different reduction methods using the non-stationary covariance function proposed in section 2.3.

The implementations of methods for all illustrations was written in Matlab and run on a processor with dual 2.8 GHz Xeon CPUs and 12GB memory. For sparse matrix calculations, we used the Matlab function *sparse*. The spam package for sparse matrix calculation in R is also available at `http://cran.rproject.org/src/contrib/PACKAGES.html`.

3.5.1 Simulation study 1

In the first simulation study, data were generated from Model (1) at 900 locations. The 800 are randomly picked over a $[0, 100] \times [0, 100]$ region and are used to estimate the parameters while the rest 100 are chosen close to the line that separates both of the regions and are used to evaluate the prediction performance.

The response $Y(s)$ is generated using model (3.1) with mean fixed to zero and non-stationary Matérn covariance structure as it is described in section 2.3. In the particular study we separate the region into two subregions with a straight line parallel to the x axis: the first subregion is defined by $y \leq 50$ and the second by $y > 50$, as it is shown in Fig. 13. Two different sets of parameters are used to evaluate the significance of the non-stationary covariance approach in the prediction.

In the first case we generate data from a stationary and isotropic Matérn co-

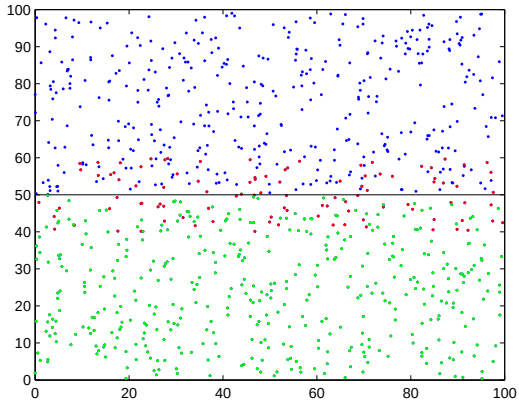


Figure 13: Spatial location of the simulated data and the 40 locations left to produce the MSPE

variance function over the whole region. We simply chose for both subregions the same parameters, with: spatial variance $\sigma^2 = 5$, nugget variance $\tau^2 = 1$, smoothing parameter $\nu = 1.5$ the range matrix parameters is $B = \Psi(0) \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix} \Psi(0)$.

In the second case we generate data from a non-stationary covariance function as it is described in section 2.3. The spatial variance σ^2, nugget variance τ^2, smoothing parameter ν are chosen the same for the two subregions and similar to the first case. The range matrix parameters for the first subregion are $B_1 = \Psi(0) \begin{pmatrix} 5 & 0 \\ 0 & 5 \end{pmatrix} \Psi(0)$ and the second subregion $B_2 = \Psi(0.5) \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix} \Psi(0.5)$.

For each set of parameters we apply four different MCMC approaches of drawing from the posterior distribution and predicting the training data. In the **first approach** a stationary covariance function is applied for the whole region and the MCMC is run to this model. In the **second approach**, the two regions are considered independent to find the posterior distributions of the stationary Matérn parameters and the prediction of the training data is done separately. In the **third approach**, the non-stationary covariance function used to generate the data is used to generate from the posterior distributions and to predict the training data. Finally, in the

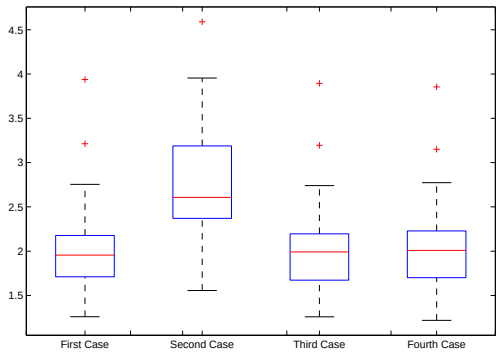


Figure 14: MSPE for four different covariance structures in the first simulation

fourth approach we find the posterior distributions separately assuming independence between the subregions but we use the non-stationary covariance matrix to predict the training data.

We simulate data thirty times with the given parameters from the stationary Matérn covariance matrix. Assuming that we separate the region into two subregions as described above, we evaluate the prediction performance for each of the approaches described above. For each simulated data and covariance structure we run the MCMC with 2000 iteretions computing also the MSPE. Thirty different means of MSPE for the four different approaches are computed and plotted in Fig. 14. The MSPE using non-stationary covariance in the MCMC algorithm is very similar to the MSPE using the stationary covariance matrix in the MCMC algorithm.

We follow the same steps as in the first case to obtain thirty different means of MSPE for the three different covariance structures. The values for the four different methods are plotted in Fig. 15.

From the above simulation study it is obvious that the non-stationary covariance function perform well in both cases. In addition, the use of the non-stationary covariance function seems to be important only in the prediction process. The posterior

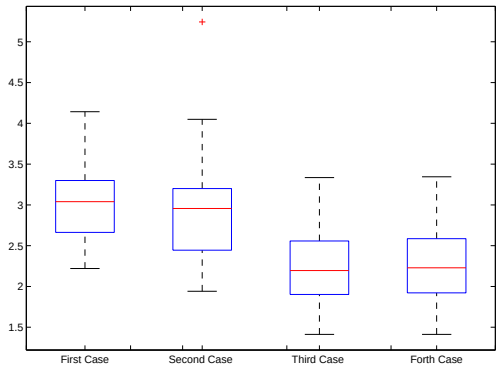


Figure 15: MSPE for four different covariance structures in the second simulation

distributions of the covariance parameters are not changing significantly if we run the MCMC separately in each subregion. This is the reason why the MSPE of the **third approach** is very similar to the MSPE of the **fourth approach**. The **third approach** will give in general slightly better results but if we have computational difficulties it is obvious we can use the **fourth approach** without really loosing in the accuracy. In the case where data are simulated from a stationary covariance function, the MSPE using non-stationary covariance in the prediction is very similar to the MSPE using the stationary covariance function. In the case where we simulate the data from a non-stationary covariance function, the use of the non-stationary covariance function in the prediction is crucial.

Moreover, the non-stationary covariance function it is crucial to be used in the case of the prediction but as we can see it is not sensitive if we compute the posterior distributions of the parameters separately.

To demonstrate the importance of the use of the non-stationary covariance matrix introduced in section 2.3 we selected 100 points in the bound 45 to 55. This was done because the difference between the non-stationary covariance and the independent subregion model are mathematically almost zero. To demonstrate this claim we

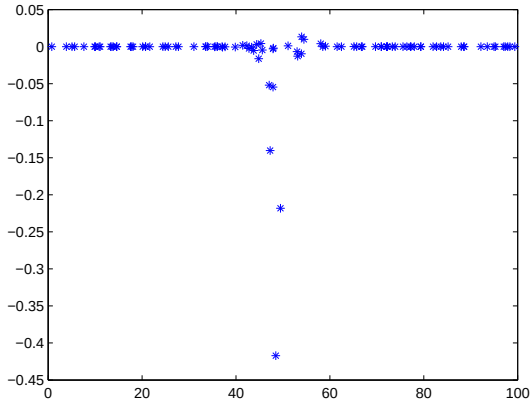


Figure 16: Difference of the absolute value of the residuals for two different methods

generate sites to evaluate the prediction performance randomly over the region. For the first case of the simulated study we repeat the same experiment 30 times and record the absolute difference of the difference of the real with the predicted value for the two different methods. (in a mathematical form $|Res^j_{non-stationary}| - |Res^j_{separate}|$). The mean of the these values is computed and plotted in Fig. 16 where it is obvious that we have differences only close to the boundary of the two regions.

3.5.2 Simulation study 2

The goal of the second simulated study is to show the effectiveness of the full rank approximation covariance function compared to predictive process and the full model when the data are generated from a Gaussian process with non-stationary covariance function. We randomly selected 2100 locations from the region $[0, 200] \times [0, 200]$ and 100 of then are left out of the study as training data. We simulated the spatial process $Y(s)$ at these 2100 locations using Gaussian process with non-stationary covariance function. The mean is modeled as $\mu(s) = 0$ for the entire region and the variance is modeled as in Section 3.2.3, partitioning the region into two different subregions with a line parallel to the x axis.

Table 4: Posterior estimations of the model parameters and the MSPE

Param.	True	Full model	$m = 200, r = 20$	$m = 200$	$m = 100, r = 20$	$m = 100$
ψ_1	0.3	0.32(0.17)	0.25(0.18)	0.56(0.22)	0.51(0.34)	0.66(0.31)
ψ_2	0.1	0.14(0.10)	0.16(0.11)	0.34(0.21)	0.16(0.10)	0.40(0.15)
λ_{11}	70	79.88(16.02)	81.63(18.81)	89.85(19.42)	85.91(22.81)	229.51(52.39)
λ_{12}	40	44.38(11.51)	49.69(12.07)	57.75(15.83)	46.73(12.74)	114.61(31.96)
λ_{21}	10	9.07(3.41)	9.57(2.90)	23.99(12.02)	14.43(5.37)	41.30(20.80)
λ_{22}	30	29.19(6.48)	35.44(8.04)	37.82(13.68)	40.27(12.9)	65.17(20.02)
σ_1^2	5	4.44(0.41)	4.20(0.45)	6.49(0.95)	4.99(0.46)	8.01(1.58)
σ_2^2	5	5.99(0.56)	5.02(0.48)	8.21(1.43)	5.18(0.52)	9.32(1.50)
τ_1^2	1	1.04(0.21)	1.06(0.20)	2.8(0.26)	1.20(0.20)	3.49(0.362)
τ_2^2	1	0.91(0.15)	0.85(0.17)	3.18(0.21)	0.83(0.17)	3.63(0.19)
MSPE	-	3.0429	3.7890	6.5644	5.032	9.740

For the Bayesian posterior inference, flat priors were assigned to each of the three intercepts, $U(0, \pi/2)$ priors were assigned for the rotation angle θ's, $U(1, d_{\max}/3)$ priors for the ϕ's, where $d_{\max}$ is the maximum distance of all pairs. The smoothness parameter ν was fixed to be 0.5 and for every subregion the variance parameters σ_k^2 are assumed to have $IG(3, 3)$ and τ_k^2 assumed to have $IG(0.5, 1)$ as priors.

For the same set of data we applied the full-scale approximation with 200 numbers of knots and 10 subpartitions in each subregion as well as the predictive process with 200 knots. Knots were located on a uniform grid over the domain. In addition to the full-scale approximation, we fit the model using the predictive process approximation with the same set of knots and the full covariance model. For each method, we ran $5,000$ iterations to collect posterior samples after a burn-in period of $1,000$ iterations. Good convergence of the respective marginal distributions is indicated by the trace plots of parameters.

Table 4 shows the Bayesian posterior sample means and standard deviations of the model parameters each approach.

In general the posterior distributions of the parameters using the full scale approximation is closer to the real posterior distribution than the posterior distributions

using the predictive process approximation. The values of the full-model and full-scale approximation are closer. Moreover the full scale approximation has a smaller MSPE.

In the next step we explore the change in the MSPE using full scale approximation and predictive process with different knots. For the same set of data we applied the full-scale approximation with a set of different knots $[100, 200, 300, 400, 500]$ numbers of knots and 10 subpartitions in each subregion as well as the predictive process with the same number of knots. For all these sets we ran $2,000$ iterations to collect posterior samples after a burn-in period of 500 iterations. For each set of parameter values, we recorded the MSPE under the four approaches for these choices of knot numbers.

3.6 Data

Stratospheric ozone is important for all life on Earth because it absorbs incoming ultra-violet (UV) radiation and also constitutes a negative radioactive forcing of climate (World Meteorological Organization, 2007, Chapter V). Since the Antarctic ozone hole was discovered in 1985 (Farman et al. (1985)), halogen-induced ozone depletion and resulting changes in atmospheric ozone distribution have been the focus of intensive research.

TOMS Level 2 data are spatially and temporally irregular measurements of Ozone following the satellite scanning tracks (measurements are 8 seconds apart) and there are a significant number of missing observations. TOMS Level 3 data are post processed from Level 2 data and they are on regular grids (1 degree latitude by 1.25 degrees longitude for pixels with latitude from 50 S to 50 N, see Krueger et al. (1998) for more details) as daily averages. Although there is loss of information in Level 3 data, especially fine scale spatial and temporal variations, data on grids with global coverage and few missing observations are convenient to focus on the study of

the covariance structure of the process purely due to the computational efficiencies.

TOMS Level 3 data are obtained usually from an ad-hoc method to average Level 2 data pixel by pixel. The main difficulty to statistically deal with the data is the computational cost. Recently, Level 2 data have been analyzed with statistical methods by Cressie and Johannesson (2008); Stein (2007b,a). Cressie and Johannesson (2008) produce new Level 3 data through statistical models rather than ad hoc averaging. The estimation of the basis function S depends on an estimation of the covariance matrix which should use repeated observations. The methods propose by Stein (2007b,a) and Jun and Stein (2008) are expensive in irregularly spaced data and are not suggested to be used in Level 2 data.

To avoid huge variances and not very good quality of data we restrict our attention to pixels with latitude from 70 S to 70 N. This is chosen also to be able to see the variation of the parameters for data that have been usually left out from other studies. To see the prediction performance of the different models, we keep out 5000 training observations at uniformly random locations in the globe. These 5000 will be used to check the performance of different methods.

3.6.1 Isotropic case

Given the subregions we consider the following model fitting methods: the full covariance model, the predictive process and the full-scale approximation. For these three methods and each subregion we use stationary Matérn covariance function with isotropic chordal distance. We remind the reader that the covariance parameter inference is done separately for each subregion. As we explained in the simulation study, this provides more accuracy and faster computational times. The estimation of the posterior parameters is not sensitive of using information from neighbor subregions when the amount of data is relatively large, e.g. $3-5$ thousand observations in each

subregions are enough to draw from the posterior with a high accuracy. This will facilitate the MCMC algorithm a lot and decrease the computational cost. On the other hand, we use two neighboring subregions for prediction purposes linking the stationary covariances with the non-stationary covariance function (3.2.3). For data prediction in the boundary of the subregion, the information from neighboring fields is significant.

In order to be able to compute and compare the posterior distribution of the full model with the approximate posterior distribution of the full-scale and predictive process, we separate the latitude space into 50 equal subregions. This will help us also to see the variation of the parameters over latitude. In the predictive process approximation we consider tree different intensities of 125, 250, 400. Knots were located on a uniform grid over each subregion domain. In the full-scale approximation, we consider the knots intensity used in the predictive process and in addition we used 40 subpartitions to capture the small scale variation as it is described in section (3.3). The additional computational cost of the full scale approximation is of order $\sum_{k=1}^{30} n_{ik}^3$ where n_{ik} is the number of the observations in the k^{th} subpartition of the i^{th} subregion. After obtaining the approximate posterior distribution for each method we conclude that the use of 125 knots will give poor fit of the data and should not be considered in practice. From a repeated study with different number of knots we prefer to use more than 250 knots for the predictive process and the full-scale approximation. In what follows we will present the study with 250 knots.

For all the three methods, we follow the same MCMC strategy. The three MCMC algorithms were run for a total of 5000 iterations and posterior inference was based on the last 4000 draws using 4^{th} moment of chain (a total of 1000 posterior draws). Competing methods can be compared based on their posterior probabilities and the mean square prediction error (MSPE) for a set of training data which have been left

out of the study. To compare the posterior probabilities for different subregions and different methods we construct the box-plots of the MCMC draws. Box-plot comparison is an easy and sufficient graphical technique to compare posterior distributions of parameters for different subregions or methods.

The box-plots of the MCMC draws for the covariance parameters $\boldsymbol{\phi}$, $\boldsymbol{\sigma}^2$ and $\boldsymbol{\tau}^2$ using the full-model are respectively presented in Fig. A-4(a),Fig. A-4(b), and Fig. A-4(c). There is a clear dependence of all the parameters on the latitude which means that the TOMS data supports the use of a non-stationary covariance matrix. We capture this non-stationary with the model described in section 2.3. For values close to the poles the variance is very big and it is not very useful to be included in the study. This is also the reason why we zoom-in the subregions with latitude close to the equator. To see how the parameters change closer to the equator we plot the MCMC box-plots posterior of $\boldsymbol{\phi}$, $\boldsymbol{\sigma}^2$ and $\boldsymbol{\tau}^2$ for 40 subregions with latitude range from -55 to 55 in Fig. A-7(a),Fig. A-7(b), and Fig. A-7(c) respectively.

We also present the MCMC posterior distributions of the covariance parameters $\boldsymbol{\phi}$, $\boldsymbol{\sigma}^2$ and $\boldsymbol{\tau}^2$, using the predictive process approximation with 250 knots in Fig. A-8(a), Fig. A-8(b) and Fig. A-8(c) and using the full-scale approximation with 250 knots and 40 sub-partitions in Fig. A-9(a), Fig. A-9(b) and Fig. A-9(c). For a complete picture, we also give the same distributions for the predictive process and the full-scale approximation from -70 to 70 in Fig. A-5 and Fig. A-6 respectively. From these graphs is obvious that the predictive process tend to overestimate all the parameters of the covariance: the range parameter, the model variance and the nugget variance. The accuracy of the predictive process with 250 knots in the model variance cannot be trusted. Instead, the posterior distributions of the parameters using the full-scale approximation are very close to the posterior distribution using the full model. We reduce the computational cost and maintain a high accuracy on

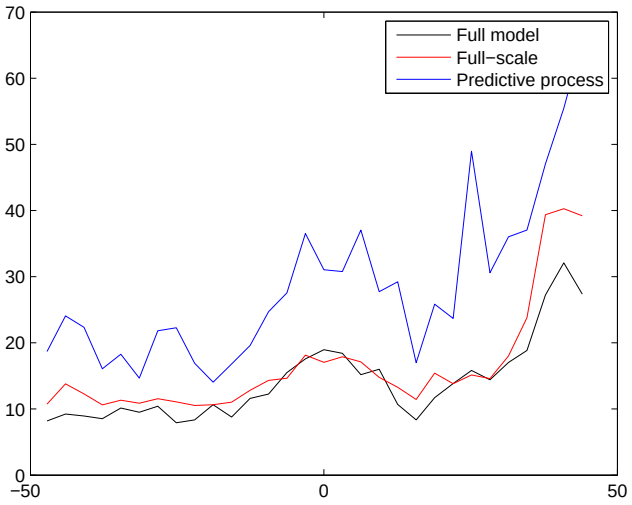


Figure 17: Comparing the MSPE for the three different methods

estimating the posterior distribution of the parameters.

To better demonstrate the accuracy of each method we test their prediction performance. We predict the values of 5000 training data using the prediction process which borrows strength from neighboring subregions as it is described in section (3.4.2). For every subregion we compute the MSPE using the predictive process approximation, the full-scale approximation and the full-model. The values of 30 subregions in the middle are given in Table 5 as well as in Fig. 17. The MSPE using the predictive process approximations is always larger than the MSPE using the full-scale approximation. As a mater of fact in some cases the MSPE using predictive process is twice as big as the MSPE using the full-scale. Moreover, the MSPE using the full scale approximation is very similar to the MSPE computed by the full-model. In general the full-scale approximation gives slightly bigger MSPE than the full-model however we have a few cases where the full-scale approximation performed better on average. This should be expected when we work with real data.

From the above study we conclude that the full-scale approximation performs

Table 5: MSPE table

Subregin	MSPE full-model	MSPE full-scale	MSPE PP
1	8.1832	10.7269	18.6851
2	9.2428	13.7994	24.0790
3	8.9502	12.2883	22.3399
4	8.5282	10.6106	16.0746
5	10.1497	11.3186	18.2895
6	9.5007	10.8723	14.6687
7	10.4195	11.5509	21.8168
8	7.9168	11.0704	22.2754
9	8.3429	10.5323	16.9125
10	10.6541	10.6388	14.0622
11	8.7700	11.0415	16.8053
12	11.5994	12.8354	19.5942
13	12.2549	14.3242	24.7129
14	15.5072	14.6327	27.5421
15	17.5676	18.1493	36.5181
16	18.9536	17.0597	31.0577
17	18.4256	17.8960	30.7692
18	15.1855	17.1268	37.0435
19	16.0168	14.7753	27.7258
20	10.6889	13.2716	29.2326
21	8.3473	11.4326	16.9738
22	11.7195	15.4102	25.8425
23	13.8429	13.8264	23.6904
24	15.8317	15.1418	48.9797
25	14.4049	14.5996	30.5840
26	17.0219	17.9945	36.0282
27	18.8360	23.8072	37.0351
28	27.2268	39.3680	47.0424
29	32.0858	40.2659	55.4432
30	26.3579	39.1945	65.7298

better than the predictive process and its results are more similar to the results produced by the full-model.

3.6.2 Anisotropic case

As we explained, in the above study we took 50 subregions to compare the full-scale and predictive process approximation with the full-model the covariance function is considered isotropic within each subregion. The next step is to introduce the anisotropy of the covariance matrix into each subregion and see whether or not the ozone data support this model. Because in the anisotropy case the distance depends also on the separation vector between locations we take thicker subregions to explore this possible dependence.

We decide to take twenty disjoint subregions in a latitude range of $[-70, 70]$. All the steps are the same with the previous study with the difference that the range parameter ϕ_i is substituted from six other parameters of the range matrix $B_i = \Psi_i \Lambda_i \Psi_i'$, $\theta_{1i}, \theta_{2i}, \theta_{3i}, \lambda_{1i}, \lambda_{2i}$ and λ_{3i}, for $i = 1, \ldots, 20$.

Since the number of the observations in some of the subregions are close to $15,000 - 20,000$ we avoid useing the full-model approximation and concentrate only on the full-scale approximation. The MCMC algorithm was run as in the isotropic case (Section (3.7.1)) with $3,000$ drwas and burn-in of 750 draws. The box-plots of the MCMC distributions of the parameters obtained using the full-scale approximation covariance with 300 knots and 40 subregionss are plotted in Fig. A-10 and Fig. A-11 for comparison. We observe that λ's, λ_1, λ_2 and λ_3, inside each subregion are different and θ's , $\theta_1, \theta_{2i}, \theta_3$, are different from zero. This means that the data supports the anisotropic model. One interesting observation is that despite having different λ's in each subregion they seems to vary across the different latitudes quite similarly.

The question of whether or not we should use the anisotropic covariance matrix

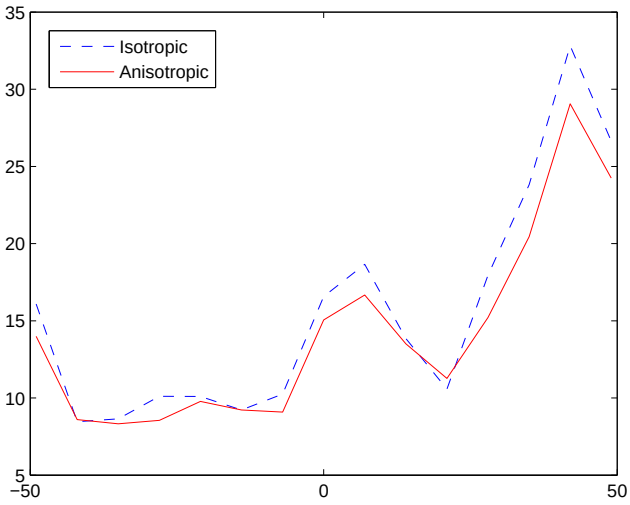


Figure 18: Comparing the MSPE of anisotropic with isotropic covariance

will depend on whether the use of the anisotropic covariance matrix will yield a smaller MSPE compared to the MSPE using the isotropic covariance matrix. Again 5000 training data are uniformly left out from the study to be used for the comparison of the prediction performance of the two different methods (covariance functions). After applying the prediction steps described in section (3.4.2) we compute the mean of the MSPE for the two different covariance functions at every single subregion and plot it in Fig. 18. The MSPE using the anisotropic covariance matrix is in general smaller than the one using isotropic covariance matrix. This shows a strong support of the ozone data to the anisotropic model.

3.7 Concluding remarks

In this chapter we focus on modeling the data in the globe with a Gaussian process. We model the mean with spherical harmonics basis and the covariance of the spatial error with a non-stationary and geometrical anisotropic closed form covariance function. We model realistically the covariance function in $3D$ in order to be able to

use the non-stationary covariance form proposed by Paciorek and Schervish (2006). The choice of this covariance function has some advantages over other possible choices. First of all the parameter inference is straightforward since the parameters are well defined. Secondly, we can partition the region to small subregions where we can infer the parameters independently from the other subregions. Third, in each subregion we can use different reduction techniques such as predictive process, tapering and the full-scale. Finally, we can partition a stationary Gaussian field without losing significantly the prediction accuracy. To deal with the large dimensionality of the data we apply the predictive process and the full-scale approximation. From the simulation study as well as the real data analysis we conclude that the performance of the full-scale approximation is closer to the full model and as such should be preferred. We also prove in practice the non-stationary and the anisotropic nature of the covariance function of TOMS level 2 data.

CHAPTER IV

MODIFIED TREED GAUSSIAN PROCESS

As it is explained in Chapter III, in the recent literature there are many proposed covariance functions for the Gaussian Process which model the non-stationary of the spatial data. The piecewise Gaussian Process (GP) is a common model for fitting non-stationary spatial data where the overall region is partitioned into smaller disjoint sub-regions with stationary Gaussian processes. The two main questions we have to answer using this model are: a) how to link the different sub-regions and b) how to separate the region into non-stationary subregions. The first question is answered in Chapter III. In this chapter we will concentrate on answering the second question.

Smith (2001) and Fuentes (2001) proposed a kernel approach in which the unknown process is taken to be the convolution of a fixed kernel over independent stationary processes, in different subregions, with different covariance parameters; Barber and Fuentes (2004) gave a discretized mixture version of the model. They suggested the use of the Akaike information criteration (AIC) to find these subregions first and then apply a Bayesian approach to find the posterior distribution in each subregion. Kim et al. (2005) used mixtures of Gaussian processes defined locally on a tessellation. Paciorek and Schervish (2006) proposed an innovative model for the covariance function which links the different pieces. In their paper the subregions are assumed known while in practice the subregions are usually unknown and not very straightforward to define.

A popular and effective method for partitioning the non-stationary region into stationary subregions is the tree Gaussian model (TGM). The use of the random number of subregions and the random boundaries of subregions make this method

very attractive for the Bayesian approach. One of the weaknesses of this method is the assumption of the independent data between subregions. In this chapter we will try to combine the Paciorek and Schervish (2006) model with the tree Gaussian model to overcome the loss of information between different subregions. Moreover, because in practice the computational cost may be very big we propose the use of the full-scale approximation technique for the covariance matrix as it is presented in the third chapter.

4.1 Bayesian inference with undefined dynamic subregions

The Bayesian treed Gaussian process (BTGP) is used in statistics to separate the space into small disjoint subregions with different parameters, see Gramacy and Lee (2008). Although the assumption of independent data between the subregions makes the BTGP attractive to deal with large dimensional data we may still have computational issues if each subregion consists of large dimensional data. TOMS data are such an example which may start with an approximately $200,000$ observations in a parent subregion. On the other hand the assumption of independence is also one of the weaknesses of the BTGP approach since it ignores possible dependence across different subregions in the grow and prune operations.

The assumption of parameters changing in latitude and not in longitude will simplify the BTGP by ignoring some computationally challenging steps. Also we simplify the BTGP by considering only split (grow), merge (prune) and change operations in the algorithm as well as the parameter updating given the subregions. We give first a review of the BTGP following Gramacy and Lee (2008) and later we propose a unique approach which fits better to our problem.

4.2 Bayesian treed Gaussian process

A tree T recursively partitions the input space into R non-overlapping regions: $r_{\xi=1}^{R}$. Each region r_ξ contains covariate and data $D_\xi = \{X_\xi, Z_\xi\}$, consisting of n_ξ observations and the generative GP model for every of the regions is:

$$Z_\xi(s) = \mu(s) + W_\xi(s) + \epsilon_\xi(s), \tag{4.1}$$

where $W_\xi(s)$ are considered independent for different region, r_ξ. While this detail is not very important when we estimate the parameters it is crucial for prediction process. Especially, for location close to the boundaries of the regions.

Following Chipman et al. (1998, 2002), the prior is specified through a tree-generating process and enforce a minimum amount of data in order to infer the parameters in each partition. Starting with a null tree (all data in a single region), a leaf node $\eta \in T$, representing a region of the input space, splits with probability $a(1 + d_\eta)^{-b}$, where d_η is the depth of $\eta \in T$ and a and b are parameters chosen to give an appropriate size and spread to the distribution of trees. Further details are available in the Chipman et al. (1998) papers and in Gramacy and Lee (2008). The prior for the splitting process involves first choosing the splitting dimension u from a discrete uniform, and then the split location ζ is chosen uniformly from a subset of the locations S in the u^{th} dimension. Integrating out dependence on the tree structure T can be accomplished via Reversible-Jump (RJ) MCMC.

Gramacy and Lee (2008) generalize the tree process by proposing to fit stationary GPs in each of the leaves of the tree but assuming independence between data of different subregions.

4.3 Modified proposed BTGP for TOMS data

Because we have separation of the space only through latitude, the general idea and the moves in the BTGP are easier. The u variable described in BTGP review is considered constant and equal to the latitude variable. Furthermore, we improve the merge and split step by using the non-stationary covariance structure to link two separated subregions. Let $\boldsymbol{\zeta}$ represent all the splitting points in the latitude direction, y dimension. Apart from the updating of the parameters given in Chapter III in this chapter we use three more different operations: split, merge and change.

The split (grow) and merge (prune) operations are complex because they add or remove partitions, changing the dimension of the parameter space. The first step for either operation is to uniformly select a (child) subregion to split (grow), or two neighbor subregions (a parent) to merge (prune). When a split move occurs, one single parent subregion creates two smaller and disjoint children subregions (or when a split move occurs one child subregion splits into two smaller disjoint subregions). New parameters must be proposed for one of the created subregions as well as for the new splitting point, ζ_{r+1}. The other children subregion absorbs its parameters by the parent subregion. In the merge (prune) operations, we randomly select parameters from one of the children subregions being absorbed.

We present in details the MCMC acceptance probability moves of split, merge and change. For all the moves we take equal probabilities: $q_{(r+1),r} = q_{r,(r+1)} = q_{r,r} = \frac{1}{3}$.

4.3.1 Prior

A tree model is identify as $(\Theta; T)$ and the usual prior assign is $\pi(\Theta; T) = \pi(\Theta|T)\pi(T)$. The prior specification of the parameters given the partitions (tree), $\pi(\Theta|T)$, is given in the third chapter. Here we specify the prior for the tree, $\pi(T)$. Depending on the approach we are taking we can specify the priors regarding the

depth of the tree or the number of subregions. We can follow Chipman et al. (1998) who's prior specification is given in details above or we can just specify the prior as a number of the subregions $\pi(T) = \pi(r)$. The distribution of $\pi(r)$ is chosen to be a truncated Poisson distribution.

4.3.2 Split operation

The first step for this operation is to uniformly select a subregion and then split it into two. To avoid very small subregions the split-point proposal distribution is chosen to be uniformly distributed in a logical range of the selected subregion. In our applications we use $U(L_j + (U_j - L_j)/6; U_j - (U_j - L_j)/6)$, where L_j and U_j are the lower and the uper latitude bound of the j^{th} subregion. Let $\Theta_j = (\phi_j, \sigma_j^2, \tau_j^2, \nu)$ denote all the parameters in the j^{th} selected subregion (parent subgegion) and $\Theta_{jk} = (\phi_{jk}, \sigma_{jk}^2, \tau_{jk}^2, \nu)$ denote the parameters of the k^{th} spit part of the j^{th} subregion (children subregions). One of the newly formed children is uniformly chosen to receive the parameters of the parent subregion. To ensure that the resulting Markov chain is ergodic and reversible, the other new sibling draws its parameters from the prior or from a distribution with similar mean with the existing one but with at lease twice the variance. In this step we increase the dimension of the parameters which describe one of the splitting parts (Θ_{jk}). Generate Θ_{jk} random variables $\Theta_{jk} = (\phi_{jk}, \sigma_{jk}^2, \tau_{jk}^2, \nu)$ from a distribution. All these new parameters are sampled from the prior distributions.

If we have r existing subregions the M-H ratio for splitting is:

$$\frac{\pi(r)}{\pi(r+1)} \frac{q_{(r+1),r}}{q_{r,(r+1)}} \frac{f(Z_{j1}, Z_{j2}|\beta, \Theta_{j1}, \Theta_{j2})\pi(\Theta_{j1})\pi(\Theta_{j2})}{f(Z_j|\beta, \Theta_j)\pi(\Theta_j)q(\Theta_{j2})}|J|,$$

where $\pi(r)$ is the prior of the number of the subregions, $q(\Theta_{j2})$ is the proposal distribution of generating parameters for one of the two formed siblings. This transformation will give a unity Jacobian term which can be ignore in the above equation.

4.3.3 Merge operation

In the merge operation we move from two neighboring subregions to one that consists of these two subregions. The first step for this operation is to uniformly sample two neighboring subregions which we are the candidates for merging. Let Θ_j and Θ_k denote the parameters in the j^{th} and k^{th} neighboring subregions (children subregion) and Θ'_h denote the proposed parameters of the united subregions. One of the selected subregions is randomly chosen to give its parameters to the new formed subregion. The other selected subregion parameters are given to a dummy variable in order to match the dimension of the parameter space. The acceptance ratio will be equal to:

$$\frac{\pi(r+1)}{\pi(r)}\frac{q_{r,(r+1)}}{q_{(r+1),r}}\frac{f(Z_j|\beta,\Theta'_h)\pi(\Theta'_h)q(\Theta_j)}{f(Z_{j1},Z_{j2}|\beta,\Theta_j,\Theta_k)\pi(\Theta_j)\pi(\Theta_j)}\frac{1}{|J|}.$$

4.3.4 Change operation

In the change operation we propose moving an existing split-point ζ_j between the two neighboring split points. The proposed values of the moving should be chosen carefully such that the subregions will maintain a certain numbers of observations and length. This is accomplished by sampling the proposed value from a narrow uniform band close to the existing ζ_j. The M-H acceptance ration for this operation is:

$$\frac{f(Z'_j,Z'_{j+1}|\beta,\Theta_j,\Theta_{j+1},\zeta'_j)}{f(Z_j,Z_{j+1}|\beta,\Theta_j,\Theta_{j+1},\zeta_j)},$$

where Z'_j and Z'_{j+1} are the observations in the new created subregions and $(Z'_j,Z'_{j+1}) = (Z_j,Z_{j+1})$.

4.3.5 Other approach

A common technique used in the BTGP to improve the acceptance ratio of the RJ-MCMC is to integrate out from the posterior distribution all the parameter which are not included in the tree T.

$$p(Z|X,T) = \int p(Z|X\Omega,T)P(T)d\Omega = \prod_{i=1}^{r} \int p(Z_i|X_i,\Omega_i)p(\Omega_i))d\Omega_i, \tag{4.2}$$

where Ω_i are all the parameters used inside of the subregin i.

In practice this can be done by sampling repeatedly the parameters of the covariance with given partition and then numerically compute an approximate solution for $\int p(Z_i|X_i,\Omega_i)p(\Omega_i)d\Omega_i$. Because of the computational cost we should and do not include too many iterations to compute this integral. Approximately 50 MCMC iterations are enough in practice to numerically have a stable and a good representation of the integral.

If we use this step we change the above acceptance rations which involve the tree movements. Set $T^{k+1} = T^*$ with probability:

$$\begin{aligned} \alpha(T^i, T^*) &= \min\Big\{\frac{q(T^*,T^i)}{q(T^i,T^*)}\frac{p(Z|X,T^*)p(T^*)}{p(Z|X,T^i)p(T^i)}\Big\} \\ &= \min\Big\{\frac{q(T^*,T^i)}{q(T^i,T^*)}\frac{\int p(Z_k|X_k,\Omega_k)p(\Omega_k)d\Omega_k}{\int p(Z_i|X_i,\Omega_i)p(\Omega_i)d\Omega_i p(T^i)}\Big\}. \end{aligned} \tag{4.3}$$

After an extensive simulation study we conclude that the differences of the two methods are minor and as such we usually prefer the less computational expensive method.

4.4 Reduction methods in the proposed BTGP

In practice the computational cost of the moves described above can be very big or even impossible. To overcome this computational challenge we apply the proposed

reduction technique of the full-scale approximation which is presented in detail in Chapter III. As we already proved in Chapter III, the full-scale approximation is a very good approximation of the real covariance matrix by capturing both the large and the small scale variation.

All the likelihoods and the acceptance rations described above should be replaced with the approximation likelihood. The accuracy of the full-scale approximation depends on the number of the knots and the range of the tapering. If all the regions have the same number of knots we may end up with a lot of subregions, since the likelihood of a better approximations is usually bigger than the approximated likelihood. If the data are considered uniformly distributed in the region then it is logical to assume that bigger subregions with a lot of data should occupy larger number of knots. Our strategy is simple. We first fix the knots and then we dynamically search for the subregions.

This is a very good and effective strategy since in reality it will give us the same degree of approximation all over the region. The split, merge and change operations will not depend on the approximation efficiency of the real distribution but on the difference in the data.

To avoid possible instability especially in the split and change operation we should consider only the cases where the number of the data and the knots are big enough.

4.5 Spatial prediction

For spatial prediction, we consider two approaches, one is the classical kriging method, i.e., the spatial best linear unbiased prediction (BLUP) given the Maximum Posterior Aposteriory (MAP) parameters, and the other is the Bayesian prediction where for every interation we conduct a prediction.

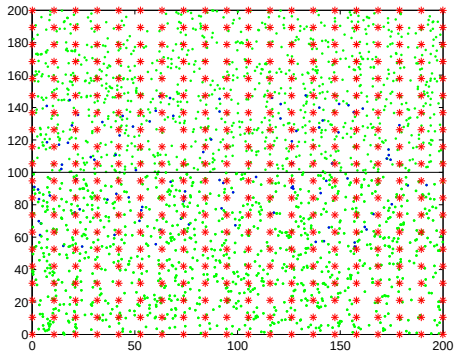


Figure 19: Simulated data and 400 knots uniformly distributed

In practice the subregions can be very big and we may not be able to store the covariance matrix which we will use in the spatial prediction. This is the reason why in practice we make the spatial prediction with a tapering form. We consider a circle with center the spatial location of the prediction .

4.6 Simulation study

The goal of this simulated study is to show the effectiveness of the proposed modified tree method to find the subregions with different parameters. We follow the procedure described in the second simulation study of Chapter III by randomly selecting 2000 locations from the region $[0, 200] \times [0, 200]$. We simulated the spatial process $Y(s)$ at these 2100 locations using the same model with the same parameters as in the third chapter. Moreover, we fix the number of knots and place them uniformly into the overall region as it is shown in the Fig. 19. The green dots represent the locations where we simulate data, the blue dots are the locations of the training data and the red stars are the location of the knots which we keep unchanged over the MCMC iterations.

We start with only one region and apply the algorithm proposed in section 4.4.

For the Bayesian posterior inference, $U(0, \pi/2)$ priors were assigned for the rotation angle θ's, $U(1, d_{\max}/3)$ priors for the ϕ's, where $d_{\max}$ is the maximum distance of all pairs. The smoothness parameter ν was fixed to be 0.5 and for every subregion the variance parameters σ_k^2 are assumed to have $IG(3, 3)$ and τ_k^2 assumed to have $IG(0.5, 1)$ as priors. We chose priors for the number of the subregions or the depth of the tree as it is described in section 4.3.1. For all the cases described below we ran 5000 MCMC iterations to collect posterior samples after a burn-in period of 1000 iterations.

4.6.1 Approximation methods in the tree process

The first question we have to answer is whether or not the use of the different reduction methods and the number of knots affect the distribution of the number of the subregions. We start with the application of the proposed algorithm using the full covariance model in order to compare it with the approximation methods. We also apply the proposed algorithm using the predictive process approximation with 169 and 400 knots and the full-scale approximation with the same number of knots and tapering range of 7 units. The prior distribution of the numbers is considered discrete uniform on $1, \ldots, 20$. The distribution of the MCMC number of the subregions for the full model is given in Fig. 20.

Moreover, the distribution of the numbers using the predictive process with 169 and 400 knots are shown in Fig. 21(a) and 21(b) respectively and the distribution of the numbers using the full scale with 169 and 400 knots and tapering range 7 are shown in Fig. 21(c) and 21(d).

Compared with the full model, all approximation approaches yield certain amount of loss in capturing the real number of the subregions, although the loss can be reduced by increasing the number of knots or the taper range. From the plots of the

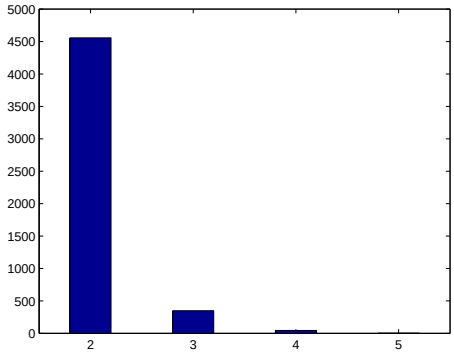


Figure 20: MCMC posterior distribution of the number of subregions when we use the full model

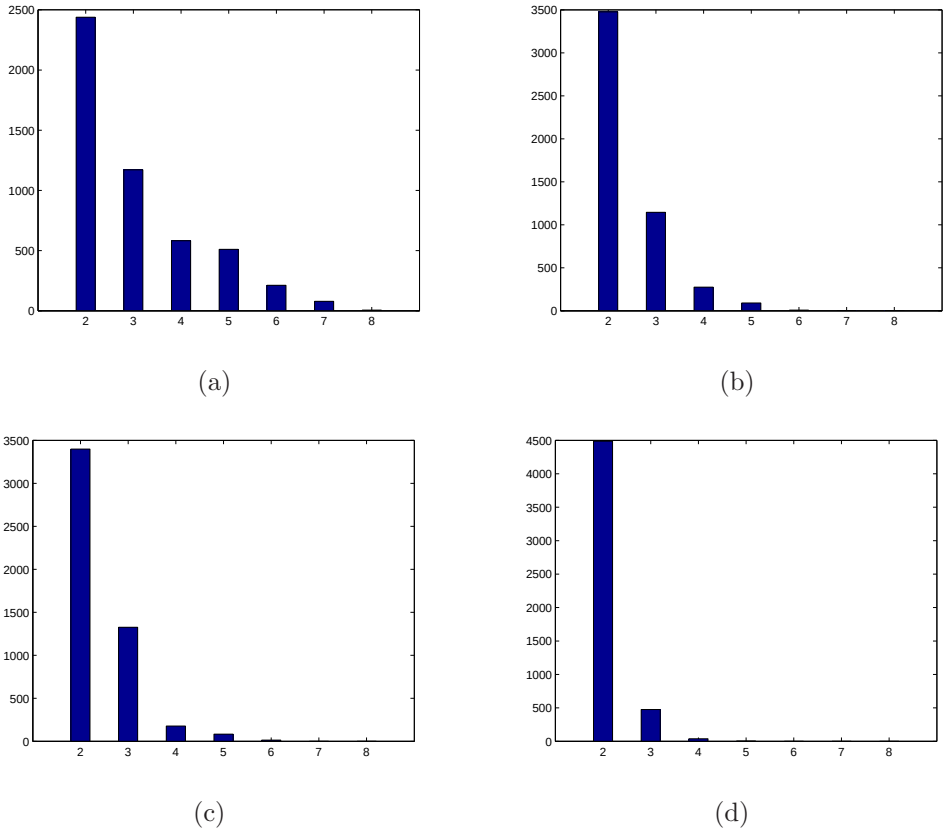


Figure 21: MCMC posterior distribution of the number of subregions: a) predictive process with 100 knots b) predictive process with 400 knots c) full-scale with 100 knots and tapering 10 d) full-scale with 400 knots and tapering 10

posterior distribution of the number of the subregions we can see that the number of knots affect the results. The predictive process with a few number of knots will usually overestimate the number of the subregions. For example, when we use 169 knots, the MCMC distribution for the number of subregions will capture the true situation in only 48.8% of the MCMC draws. This is a weak performance not only compared to the posterior of the subregions using the full model, but also to the posterior of the subregions using the full-scale approximation with same number of knots. When we increase the number of the knots to 400, we observe an increase of almost 20% in capturing the real situation but it is still very ineffective compared to the full-scale approximation with the same number of knots. The predictive process overestimates the number of the subregions even when the number of knots is 400. One important observation is that the use of the full-scale approximation with 400 knots is very similar to the full model. The improvements of the posterior distribution of parameters are minor when we take more than 400 knots.

In general, the full-scale approximation should be preferred over predictive process since the last one is quite sensitive to the choice of the number of knots.

4.6.2 Full-scale approximation in the tree process

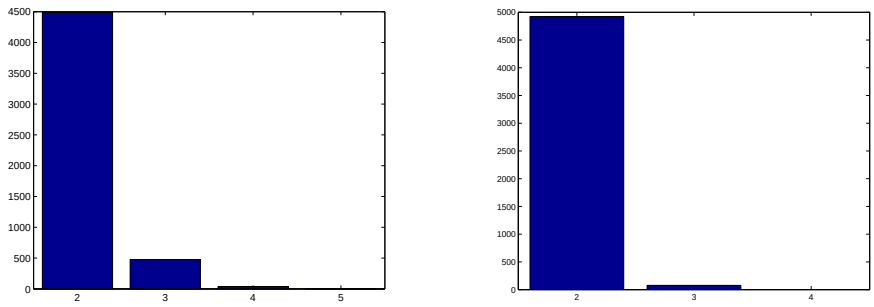
We now examine how the full-scale approximation can be improved from the priors and compare it with other models which also use the full-scale approximation.

Two priors are chosen for the number of subregions: a) the discrete uniform on $[1, \ldots, 20]$ and b) the truncated $Poisson(3)$ for values $[1, \ldots, 20]$. In the case where we take uniform distribution prior for the number from the 5000 iterations: 4488 have two subregions, 476 have three subregions, 39 have four subregions and 3 have five subregions. In the case where we take the truncated $Poisson$ prior for the number of subregions, from the 5000 iterations, 4959 have two subregions and only 41

Table 6: Posterior estimation and the MSPE for two subregions

Object	True values	$m = 400$, $r = 10$
ψ_1	0.3	0.40(0.092)
ψ_2	0.1	0.11(0.063)
λ_{11}	70	59.88(15.361)
λ_{12}	40	42.42(12.204)
λ_{21}	10	12.64(-)
λ_{22}	30	38.96(10.808)
σ^2	5	4.49(0.459)
τ^2	1	1.06(0.172)
MSPE	-	4.350

iterations have three subregions. The use of a penalty in the number of the subregions improves the MCMC percentage to capture the true number of the subregions. The MCMC distribution of the number of subregions for these two different cases is given in Fig. 22.



(a) (b)

Figure 22: MCMC posterior distribution of the number of subregions: a) uniform prior and b) Poission prior

More than 85% of the splits given that we had two or more subregions occur in the first subregion within $y \in [0, 100]$. This is suggesting that different values of the range matrix give more stable TGP than others.

To see whether or not the partitions are correct we have to explore also if the

Table 7: MSPE for four different methods

Method	MSPE
Correct Partions Anisotropic	3.659
Correct Partions Isotropic	5.387
Wrong Fixed Partions	8.296
Unknown Random	4.350

values of $\boldsymbol{\zeta}$ are close to the real one. For this reason when we have only two partitions we take the values of ζ. Also, 5 first iterations are taken out every time we jump from three to two subregions. The distribution of these values is given with a histogram in Fig. 23.

These MCMC values of ζ are very close to 100 which is our splitting point. The MCMC mean splitting point is equal to 99.7 which can be seen as a small bias. This bias seems to be consistent also for other simulations and has to be investigated.

The mean values of the posterior parameters and their standard deviation, for the above case, are given in Table 6. The mean values are relatively close to the mean values computed with fixed and known partitions. A very important observation here is that the values of the first subregion seems to be underestimated. This can be related to the bias of ζ we described above.

From the above analysis it is obvious that the proposed method is a good choice to separate the region into stationary subregions.

Finally, to demonstrate the success of the proposed approach we compare the MSPE of different approaches using the same data set. We consider four different cases: a) correct partitions with anisotropic covariance matrix b) correct partitions with isotropic covariance matrix c) wrong partitions with anisotropic covariance matrix (we take three different equal areas subregions) and d) unknown partitions with anisotropic covariance matrix.

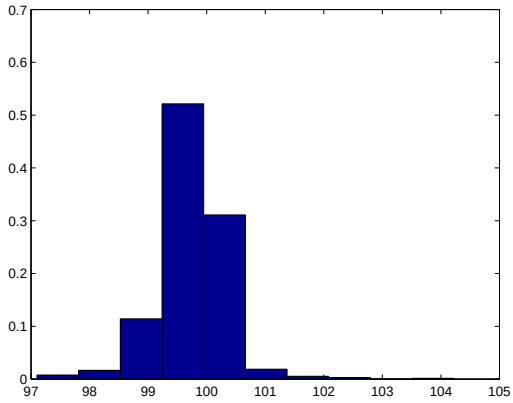


Figure 23: Histogram of the MCMC values of ζ when the number of subregions is two

The MSPE of these different approaches is given in table 7. The case of known partitions with non-stationary covariance function will give us the smallest MSPE while the uses of wrong subregions can even double the MSPE. Using the proposed tree model approach is slightly worse than the case where we know the subregion and better than the cases of wrong subregions. If we are not sure about the right subregions of the data, it is better to approach the problem with the proposed method.

4.7 Real data analysis

Stratospheric ozone data which are analyzed in Chapter III are also the data used in this chapter with the difference that the number of subregions is considered unknown and random. Stationary covariance functions are used in each subregion.

Because of the computational cost, we start the analysis by using 50 different partitions as it was done in the previous chapter. We select 7000 knots uniformly distributed in the globe with latitude rage from 70^o S to 70^o N and keep them fixed in every MCMC iteration (in which the subregions may change). Each subregion in the MCMC iterations has a different number of knots which depends on the area it occupy.

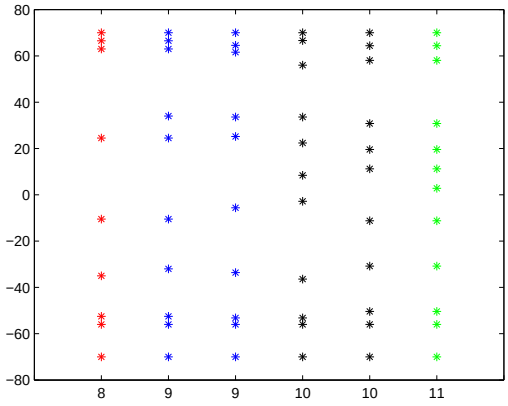


Figure 24: Six different partitions in the tree process

Subregions with a larger area have a bigger number of knots. In addition, to avoid small and unstable subregions, we apply some restrictions in the MCMC algorithm when we chose them. Following an explanatory study, which is done without any restriction, we decide to take only subregions with at least 1000 observations and 100 knots. These two restrictions avoid also numerical instability which can arise from the approximation techniques.

By applying the method described in section 4.6 we conclude that the number of the subregions is between 8 to 11. More explicitly, after 5000 iterations and a burning period of 1000 iterations we have approximately: 6% of the MCMC sample with 8 subregions, 68% of the MCMC sample with 9 subregions, 23% of the MCMC sample with 10 subregions and 2% of the MCMC sample with 11 subregions. Six of the MCMC subregions are plotted in Fig. 24 for a better understanding of the partition of the space. As we can see from this figure the non-stationarity seems to be more severing close to the poles since we have a larger and constant number of subregions. Especially between the latitude bound 60 to 70 we have three distinct subregions and in -70 to -55 we have two.

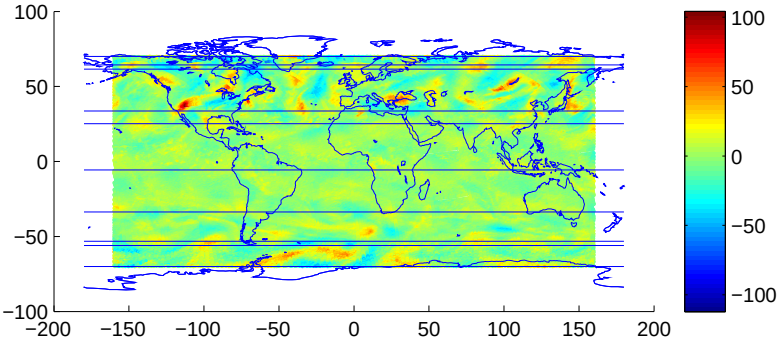
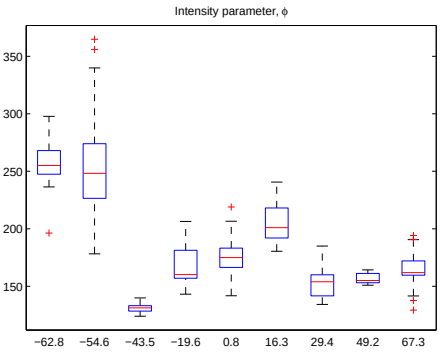
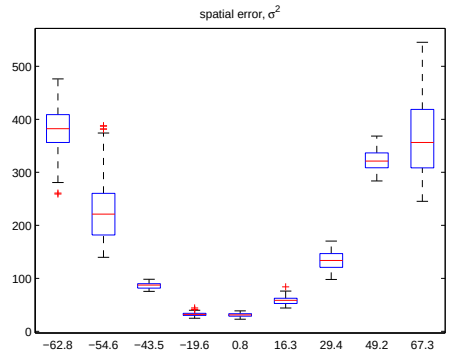


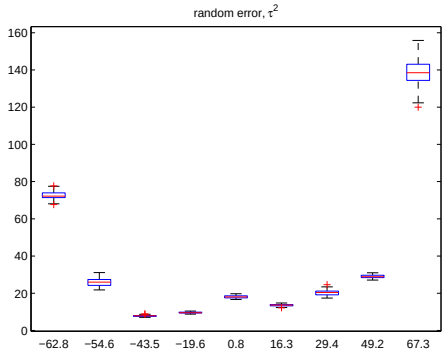
Figure 25: MAP estimation of the Bayesian treed GP



(a) Range parameter ϕ



(b) Variance of model σ^2



(c) Variance of error τ^2

Figure 26: Posterior distribution of the parameters using full-scale approximation

We compute the Maximum A Posteriori (MAP) estimates of the parameters for these 5000 iterations and we plot the MAP partition in Fig. 25. Given the subregions where we take the MAP values for the distribution we run 5000 MCMC iterations with fixed partitions as it is described in the third chapter. The distribution of these parameters are given in Fig. 26.

4.8 Concluding remarks

We developed a Bayesian treed GP model for TOMS data which can be extended to any other non-stationary data set. To make it computationally feasible, we apply reduction techniques to simplify the different MCMC operations. Moreover, we improve the prune, split and change operation by considering dependence between subregions. The method can be seen as a simple and efficient way of modeling and computationally dealing with nonstationary and big dimensional data sets.

More specifically, we partitioned dynamically the space into smaller subregions with similar covariance structure which can be linked with the non-stationary covariance function proposed by Paciorek and Schervish (2006). To deal with the high dimensionality we used the predictive process and full-scale approximation. An extensive comparison of the performance between two reduction techniques with different number of knots and tapering range is done in the simulation study where we prove the efficiency of using the full-scale approximation. We apply the proposed algorithm to TOMS data where we assumed only partitions in latitude. We consider this a successful application since the number of the subregions converges into a stable number of the subregions.

CHAPTER V

CONCLUSION

We have developed two different models for images and global spatial processes in the Bayesian paradigm using various techniques to handle the high dimensionality. Both the image pixels, which are analyzed in Chapter II, and TOMS data, which are analyzed in Chapters III and IV, are high dimensional and spatially correlated.

In Chapter II we model realistically different geometrical objects in an image. A marked point process is developed and a hierarchical Bayesian model is used to link different components of different mathematical models. More specifically, we treated the objects in the image as a known shape, wherein the geometrical properties are largely determined by templates and the interaction between the objects was modeled using the area interaction process prior (AIPP). In addition, we model the covariance structure of the likelihood with a CAR model to facilitate the computations. Finally, to solve the intractability of the posterior distribution we proposed a complex Markov Chain Monte Carlo (MCMC) algorithm which involves Reversible Jump, Metropolis-Hasting, Gibbs sampling and a Monte Carlo Metropolis-Hastings (MCMH) for the intractable normalizing constants in the prior. We successfully applied this algorithm to real TEM images, to find the characteristics of the nanoparticles, outperforming convention tools aided by manual screening.

In Chapter III we model the global data with a Gaussian process (GP) in three dimensional spaces which has a non-stationary covariance function in latitude. The use of the three dimensional chordal distance helped us to produce covariance functions which are geometrically anisotropic within each subregion and also to use well established and more general non-stationary covariance matrices given in Paciorek

and Schervish (2006). The use of the Paciorek and Schervish (2006) non-stationary covariance structure helped us to reduce the computational in two ways: a) to apply different approximation techniques for the covariance matrix and b) carry out the computations of the parameters separately for each subregion. We applied three different covariance approximation techniques: a) the predictive process, b) tapering and c) the full-scale approximation and proved with simulation studies and with real data analysis that the full-scale covariance approximation is the best reduction technique regarding the MSPE. The predictive process approximation is proven to give biased posterior distributions while the full-scale approximation is relatively unbiased. More explicitly, we demonstrate that the predictive process will always overestimate the nugget variance, the model variance and the number of the subregions in a dynamic system.

In Chapter IV we extend the application of the reduction techniques from non-stationary Gaussian process (GP) with known subregions to non-stationary GP with unknown subregions. Bayesian treed GP is used to model the covariance function dynamically and find the unknown subregions. We used the predictive process and the full-scale approximation to facilitate the MCMC operations which in practice can be computationally impossible to carry out. The predictive process approximation is proven to overestimate the number of the subregions while the full-scale approximation gives very similar results to the full model. Finally, we improve the operations in the existing dynamic model with the use of the dependent subregions.

REFERENCES

Al-Awadhi, F., Hurn, M.A., Jennison, C., 2004a. Improving the acceptance rate of reversible jump mcmc proposals. Statistics and Probability Letters 69, 189–198.

Al-Awadhi, F., Jennison, C., Hurn, M.A., 2004b. Statistical image analysis for a confocal microscopy two-dimensional section of cartilage growth. Journal of the Royal Statistical Society, Series C 53, 31–49.

Anderes, E.B., Stein, M.L., 2008. Estimating deformations of isotropic Gaussian random fields on the plane. Annals of Statistics 36, 719–741.

Arfken, G., 1970. Bispherical Coordinates (ξ, η, ϕ). 2nd edition. Academic Press Inc, Orlando, FL.

Baddeley, A.J., van Lieshout, M.N.M., 1993. Stochastic geometry models in high-level vision. Journal of Applied Statistics 20, 231–256.

Baker, C., 1977. The Numerical Treatment of Integral Equations. Clarendon, Oxford, UK.

Banerjee, S., Carlin, B., Gelfand, A., 2004. Hierarchical Modeling and Analysis for Spatial Data. Chapman & Hall-CRC, Boca Raton, FL.

Banerjee, S., Gelfand, A., Finley, A., Sang, H., 2008. Gaussian predictive process models for large spatial data sets. Journal of the Royal Statistical Society, Series B 70, 825–848.

Barber, J.J., Fuentes, M., 2004. Nonstationary spatial process modeling of atmo-

spheric pollution data. Technical Report. Department of Statistics, North Carolina State University, Raleighe.

Berger, J.O., Oliveira, V.D., Sanso, B., 2001. Objective Bayesian analysis of spatially correlated data. Journal of the American Statistical Association 96, 1361–1374.

Blake, A., Yuille, A., 1992. Active Vision. MIT Press, Cambridge, MA.

Calder, C., 2008. A dynamic process convolution approach to modeling ambient particulate matter concentrations. Environmetrics 19, 39–48.

Chen, M.H., Shao, Q.M., 1998. Monte Carlo methods on Bayesian analysis of constrained parameter problems with normalizing constants. Biometrika 85, 73–87.

Chipman, H., George, E., McCulloch, R., 1998. Bayesian CART model search. Journal of the American Statistical Association 93, 935–960.

Chipman, H., George, E., McCulloch, R., 2002. Bayesian treed models. Machine Learning 48, 303–324.

Cressie, N., 1993. Statistics for Spatial Data. 2nd edition. John Wiley and Sons Inc, New York.

Cressie, N., Johannesson, G., 2008. Fixed rank kriging for very large spatial data sets. Journal of the Royal Statistical Society, Series B 70, 209–226.

Denison, D., Mallick, B., Smith, A., 1998. A Bayesian CART algorithm. Biometrika 85, 363–377.

Dryden, I.L., Mardia, K.V., 1998. Statistical Shape analysis. 2nd edition. John Wiley and Sons Inc., Chichester.

El-Sayed, M.A., 2001. Some interesting properties of metals confined in time and nanometer space of different shapes. Accounts of Chemical Research 34, 257–264.

Farman, J.C., Gardiner, B.G., D.Shanklin, J., 1985. Large losses of total ozone in Antartica reveal seasonal ClOx/NOx interaction. Nature 315, 207–210.

Finley, A., Sang, H., Banerjee, S., Gelfand, A., 2009. Improving the performance of predictive process modeling for large datasets. Computational Statistics and Data Analysis 53, 2873–2884.

Fuentes, M., 2001. A high frequency kriging approach for non-stationary environmental processes. Environmetrics 12, 469–483.

Fuentes, M., 2002. Spectral methods for nonstationary spatial processes. Biometrika 89, 197–210.

Gelman, A., Carlin, J., Stern, H., Rubin, D., 2004. Bayesian Data Analysis. 2nd edition. Chapman & Hall, Boca Raton, FL.

Genton, R.F.M., Nychka, D., 2006. Covariance tapering for interpolation of large spatial datasets. Journal of Computational and Graphical Statistics 15, 502–523.

Geyer, C.J., Møller, J., 1994. Simulation procedures and likelihood inference for spatial point processes. Scandinavian Journal of Statistics 21, 359–373.

Geyer, C.J., Thompson, E.A., 1995. Annealing Markov chain Monte Carlo with applications to ancestral inference. Journal of the American Statistical Association 90, 909–920.

Gneiting, T., 2002. Compactly supported correlation functions. Journal of Multivariate Analysis 83, 493–508.

Gramacy, R.B., Lee, H.K.H., 2008. Bayesian treed Gaussian process models with an application to computer modeling. Annals of Statistics 103, 1119–1130.

Green, P., 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika 82, 711–732.

Grenander, U., 1993. Statistical Shape Analysis. 2nd edition. Oxford University Press, Oxford, UK.

Grenander, U., Miller, M.I., 1995. Representations of knowledge in complex systems. Journal of the Royal Statistical Society, Series B 56, 549–603.

Helterbrand, J.D., Cressie, N., Davidson, J.L., 1994. A statistical approach to identifying closed object boundaries in images. Advances in Applied Probability 26, 831–854.

Higdon, D., 1998. A process-convolution approach to modeling temperatures in the north Atlantic Ocean. Journal of Environmental and Ecological Statistics 5, 173–190.

Higdon, D.M., Swall, J., Kern, J.C., 2011. Non-stationary spatial modeling. In Bayesian statistics 6: Proceedings of the sixth Valencia International Meeting, Oxford University Press, UK.

Hobolth, A., Kent, J.T., Dryden, I.L., 2002. On the relation between edge and vertex modeling in shape analyis. Scandinavian Journal of Statistics 29, 355–374.

Horn, R., Johnoson, C., 1985. Matrix Analysis. Cambridge University Press, MA, Cambridge.

ImageJ, 2004. Image processing and analysis in java. `http://rsbweb.nih.gov/ij/`. [Online; accessed 1-January-2011].

Johannesson, G., Cressie, N., H.-C. Huang, H., 2007. Dynamic multi-resolution spatial models. Environmental and Ecological Statistics 14, 5–25.

Jun, M., Stein, M.L., 2008. Nonstationary covariance models for global data. Annals of Applied Statistics 2, 1271–1289.

Jung, M.R., Ko, J.H.S.B., Nam, J.Y., 2008. Automatic cell segmentation and classification using morphological features and Bayesian networks. The Proceedings of the Society of Photo-Optical Instrumentation Engineers 6813, 68130.

Kaufman, C., Schervish, M., Nychka, D., 2008. Covariance tapering for likelihood-based estimation in large spatial data sets. Journal of the American Statistical Association 103, 1545–1555.

Kim, H.M., Mallick, B., Holmes, C., 2005. Analyzing nonstationary spatial data using a piecewise Gaussian process. Journal of the American Statistical Association 470, 653–658.

Kothari, S., Chaudhry, Q., Wang, M., 2009. Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques. IEEE International Symposium on Biomedical Imaging 1, 795–798.

Krueger, A.J., Bhartia, P.K., McPeters, R.D., Herman, J.R., Wellemeyer, C.G., Jaross, G., Seftor, C.J., Torres, O., Labow, G., Byerly, W., Taylor, S.L., Swissler, T., Cebula, R.P., 1998. Adeos total ozone mapping spectrometer (TOMS) data products users guide. `http://toms.gsfc.nasa.gov/datainfo/adeos_userguide.pdf`.

Liang, F., 2010. A double Metropolis-Hastings sampler for spatial models with in-

tractable normalizing constants. Journal of Statistical Computing and Simulation 80, 1007–1022.

Liang, F., Jin, I.H., 2011. A Monte Carlo Metropolis-Hastings algorithm for sampling from distributions with intractable normalizing constants. Technical Report. Department of Statistics, Texas A&M University, College Station.

van Lieshout, M.N.M., 2008. Depth map calculation for a variable number of moving objects using Markov sequential object processe. IEEE, Transactions on Pattern Analysis and Machine Intelligence 30, 1308–1312.

Mardia, K.V., Qian, W., Shah, D., de Souza, K.M.A., 1997. Deformable template recognition of multiple occluded objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 1035–1042.

Mohamed, M.B., Volkov, V., Link, S., El-Sayed, M.A., 2000. Lightning gold nanorods: Fluorescence enhancement of over a million compared to the gold metal. Chemical Physics Letters 317, 517–523.

Møller, J., Pettitt, A.N., Reeves, R., Berthelsen, K.K., 2006. An efficient Markov chain Monte Carlo method for distributions with intractable normalizing constants. Biometrika 93, 451–458.

Murray, I., Ghahramani, Z., MacKay, D., 2006. MCMC for doubly-intractable distributions. Proc. 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI), Cambridge, MA .

Nehl, C.L., Liao, H., Hafner, J.H., 2006. Optical properties of star-shaped gold nanoparticles. Nano Letters 6, 683–688.

Nychka, D., Royle, C.W.K.A., 2002. Multiresolution models for nonstationary spatial covariance functions. Statistical Modelling 2, 315–332.

Paciorek, C., Schervish, M., 2006. Spatial modelling using a new class of nonstationary covariance functions. Environmetrics 17, 483–506.

Pan, Y., Neuss, S., Leifert, A., Fischler, M., Wen, F., Simon, U., Brandau, M.S.W., Jahnen-Dechent, W., 2007. Size-dependent cytotoxicity of gold nanoparticles. Small 3, 1941–1949.

Pievatolo, A., Green, P.J., 1998. Boundary detection through dynamic polygons. Journal of the Royal Statistical Society, Series B 60, 609–626.

Qian, W., Mardia, K., 1995. Recognition of multiple objects with occlusions. Technical Report. Department of Statistics, University of Leeds, UK.

Rue, H., Hurn, M.A., 1999. Bayesian object identification. Biometrika 86, 649–660.

Rue, H., Syversveen, A., 1998. Bayesian object recognition with baddeley's delta loss. Advances in Applied Probability 30, 64–84.

Sampson, P.D., Guttorp, P., 1992. Nonparametric estimation of nonstationary spatial covariance structure. Journal of the American Statistical Association 87, 108–119.

Sang, H., Huang, J.Z., 2011. A full-scale approximation of covariance functions for large spatial data sets. Journal of the Royal Statistical Society, Series B, In press.

Schabenberger, O., Gotway, C., 2005. Statistical Methods for Spatial Data Analysis. 2nd edition. Chapman & Hall, Boca Raton, FL.

Schmidt, A.M., O'Hagan, A., 2003. Bayesian inference for non-stationary spatial covariance structure via spatial deformations. Journal of the Royal Statistical Society, Series B 65, 743–758.

Smith, R., 2001. Environmental statistics. Technical Report. Department of Statistics, University of North Carolina, Chapel Hill.

Stein, M.L., 1999. Interpolation of Spatial Data: Some Theory for Kriging. 2nd edition. Springer, New York.

Stein, M.L., 2007a. Seasonal variations in the spatial-temporal dependence of total column ozone. Environmetrics 18, 71–86.

Stein, M.L., 2007b. Spatial variation of total column ozone on a global scale. Annals of Applied Statistics I, 191–210.

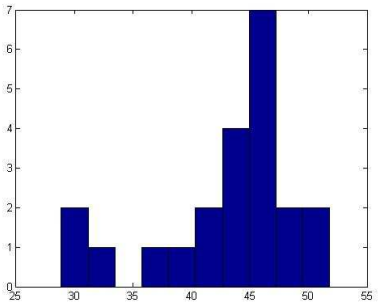
Stein, M.L., 2008. A modeling approach for large spatial datasets. Journal of the Korean Statistical Society 37, 3–10.

Wang, Z.L., Petroski, J.M., Green, T.C., El-Sayed, M.A., 1998. Shape transformation and surface melting of cubic and tetrahedral platinum nanocrystals. Journal of Physical Chemistry B 102, 6145–6151.

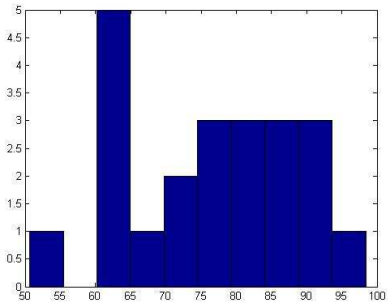
Williams, C., Seeger, M., 2001. Using the Nyström method to speed up kernel machines. Advances in Neural Information Processing Systems 13, 682–688.

APPENDIX A

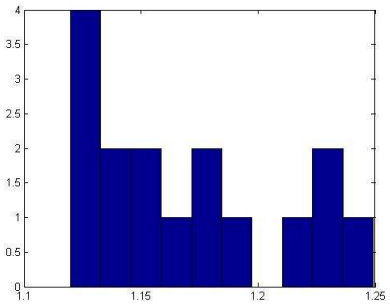
ADDITIONAL FIGURES ANALYSED IN CHAPTERS II AND III



(a) s(scale)

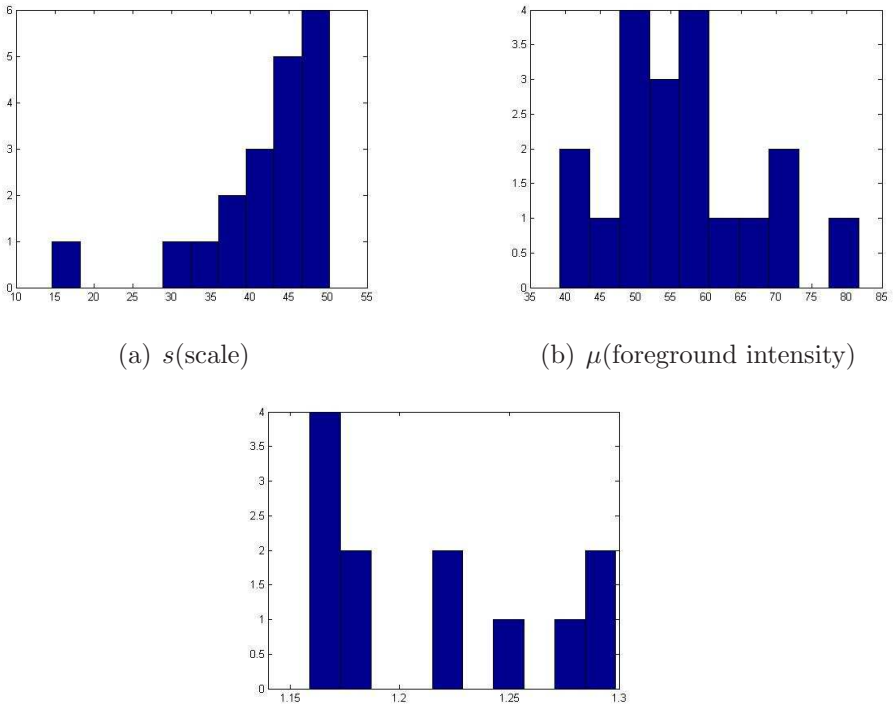


(b) μ(foreground intensity)



(c) g^r(random pure parameter)

Figure A-1: Distribution of the MAP estimates for some shape parameters in Ex1

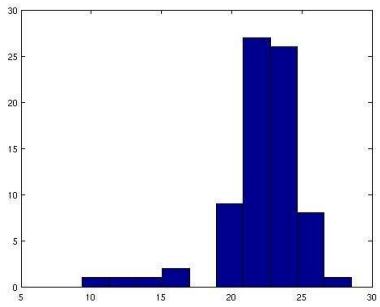


(a) s(scale)

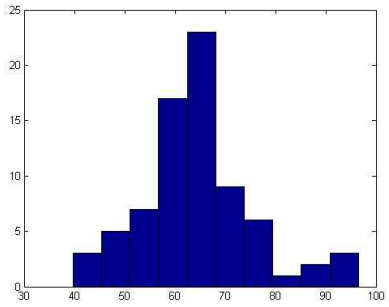
(b) μ(foreground intensity)

(c) g^r(random pure parameter)

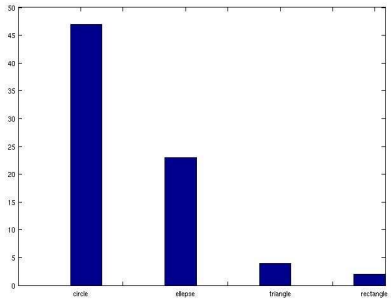
Figure A-2: Distribution of the MAP estimates for some shape parameters in Ex2



(a) s(scale)

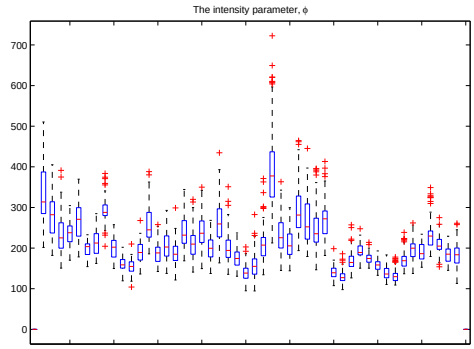


(b) μ(foreground intensity)

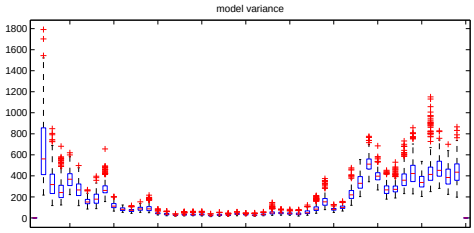


(c) g^r(random pure parameter)

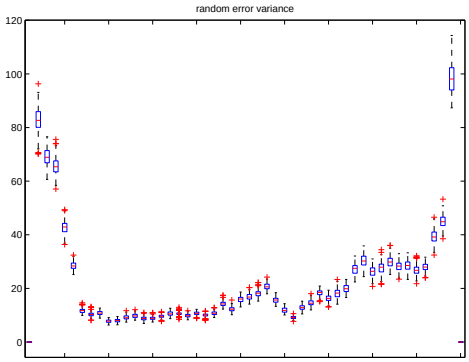
Figure A-3: Distribution of the MAP estimates for some shape parameters in Ex3



(a) Range parameter ϕ

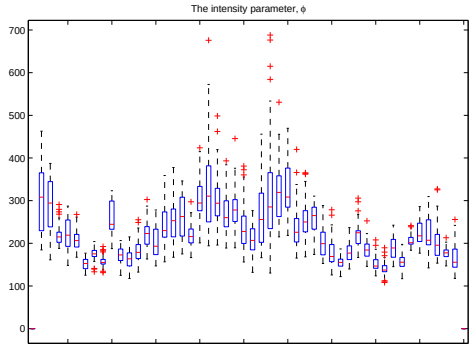


(b) Variance parameter σ^2

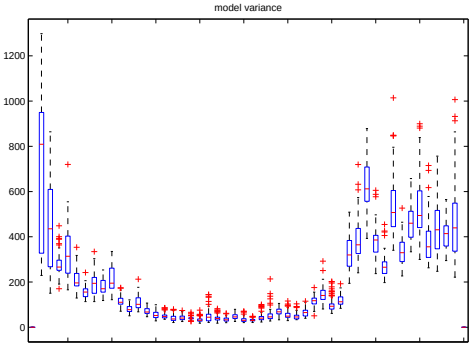


(c) Variance of error parameter τ^2

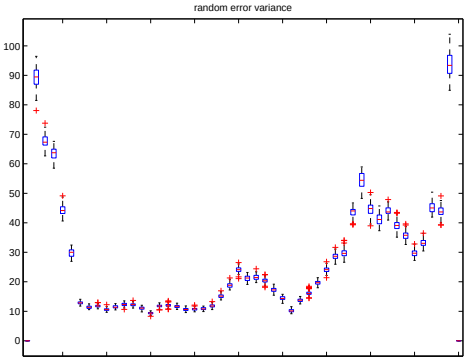
Figure A-4: Posterior distribution of the parameters using full model for 50 different bands in the latitude range [-70,70]



(a) Range parameter ϕ

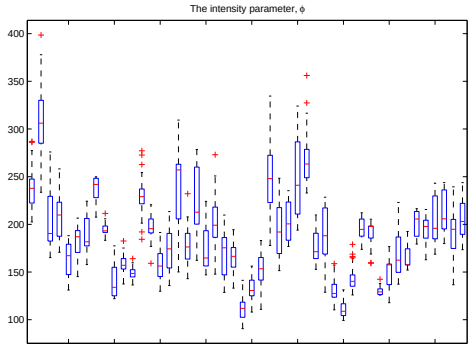


(b) Variance parameter τ^2

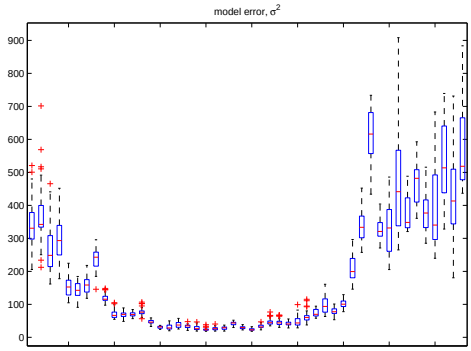


(c) Variance of error parameter σ^2

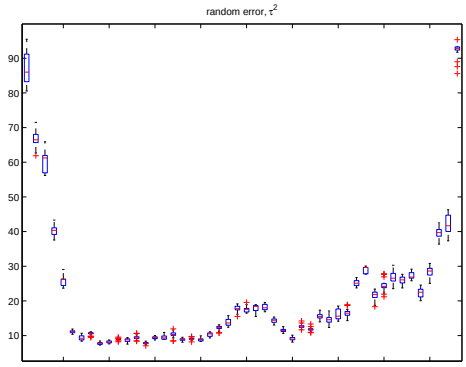
Figure A-5: Posterior distribution of the parameters using the predictive process for different band in the latitude range [-70,70] using 250 knots



(a) Range parameter ϕ

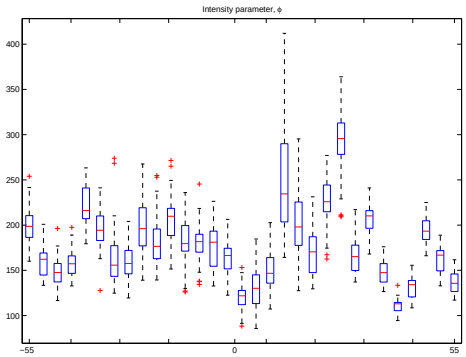


(b) Variance parameter σ^2

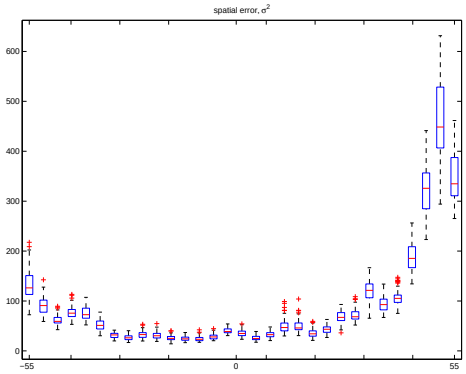


(c) Variance of error τ^2

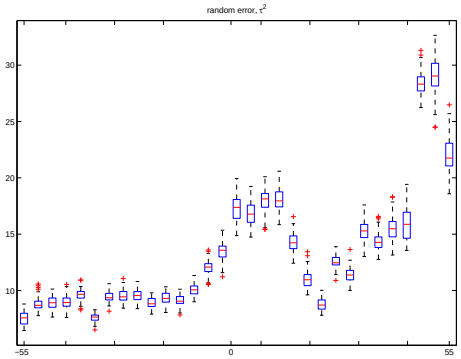
Figure A-6: Posterior distribution of the parameters using full rank approximation for different bands in the latitude range of [-70,70] with 250 knots and 30 subregions



(a) Range parameter ϕ

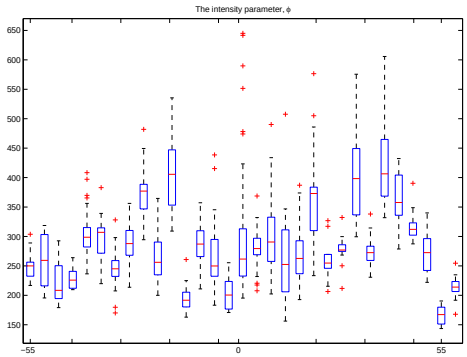


(b) Variance parameter σ^2

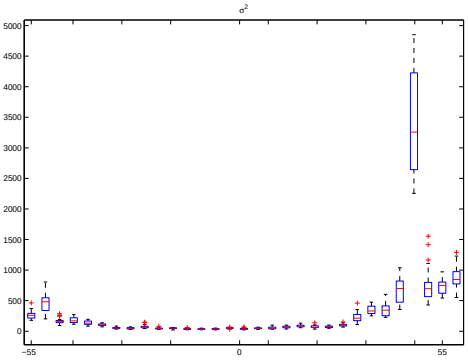


(c) Variance of error parameter τ^2

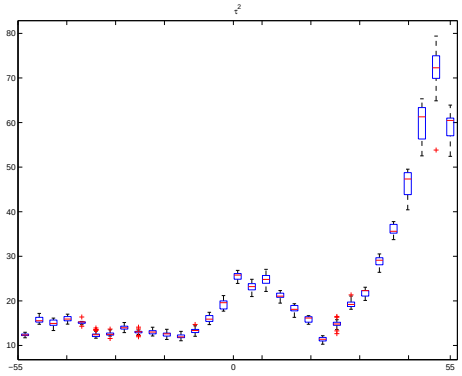
Figure A-7: Posterior distribution of the parameters using the full model for different bands in the latitude range of [-55,55]



(a) Range parameter ϕ

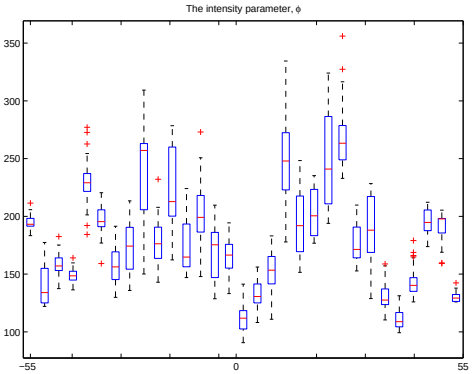


(b) Variance parameter σ^2

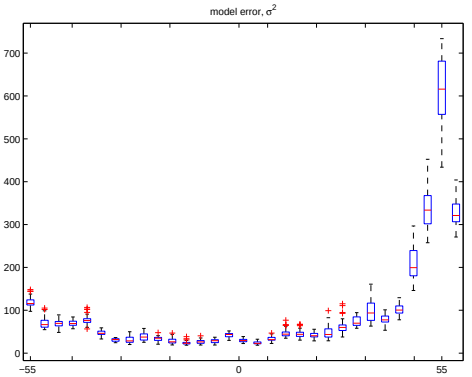


(c) Variance of error τ^2

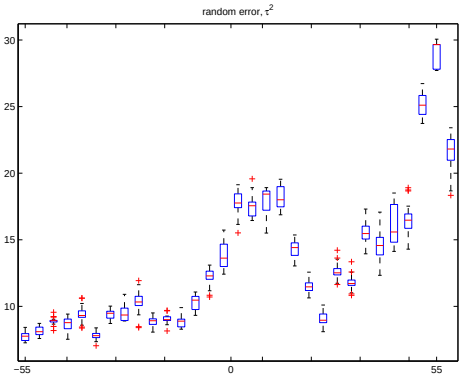
Figure A-8: Predictive proccess boxplot for different bands in the rage of latitude [-55,55] using 250 knots



(a) Range parameter ϕ

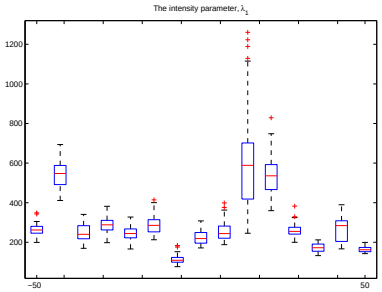


(b) Variance of model σ^2

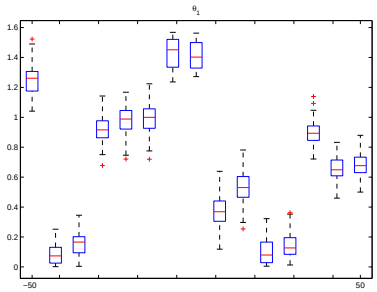


(c) Variance of error τ^2

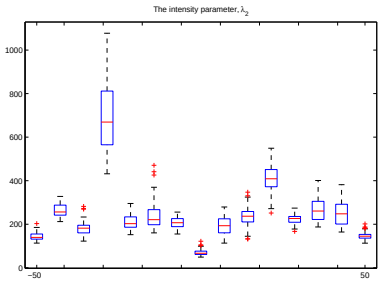
Figure A-9: Posterior distribution of the parameters using full-scale approximation with 250 knots and 30 subregions



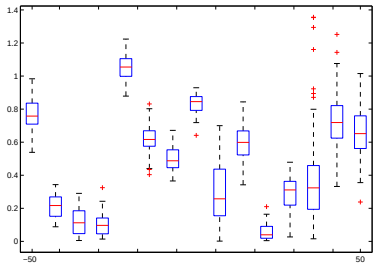
(a) λ_1



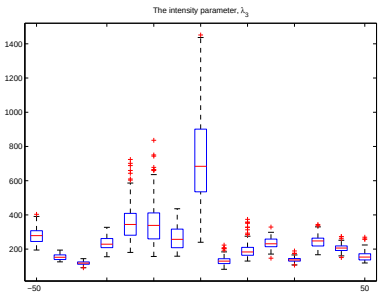
(b) θ_1



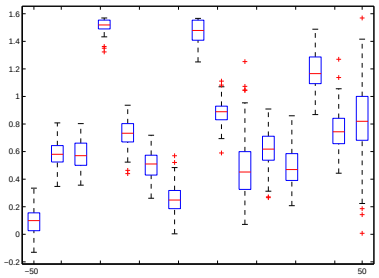
(c) λ_2



(d) θ_2

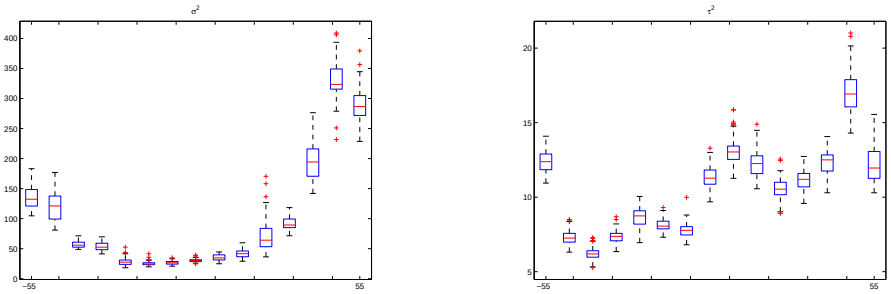


(e) λ_3



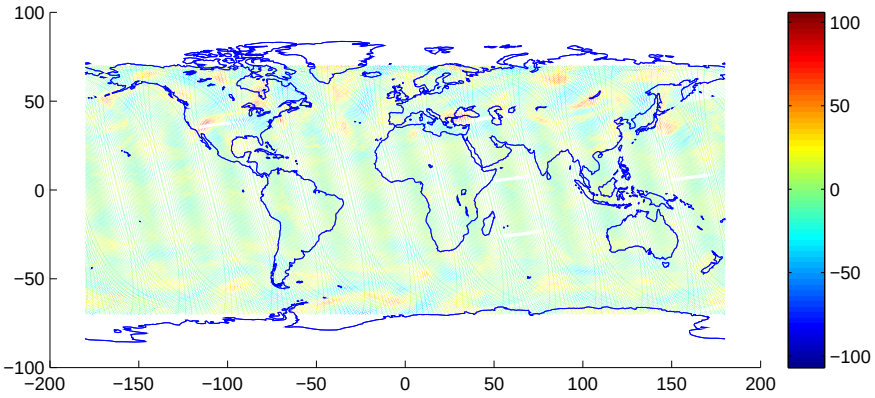
(f) θ_3

Figure A-10: The posterior distribution of the parameter of the "Range" matrix in the 3D model

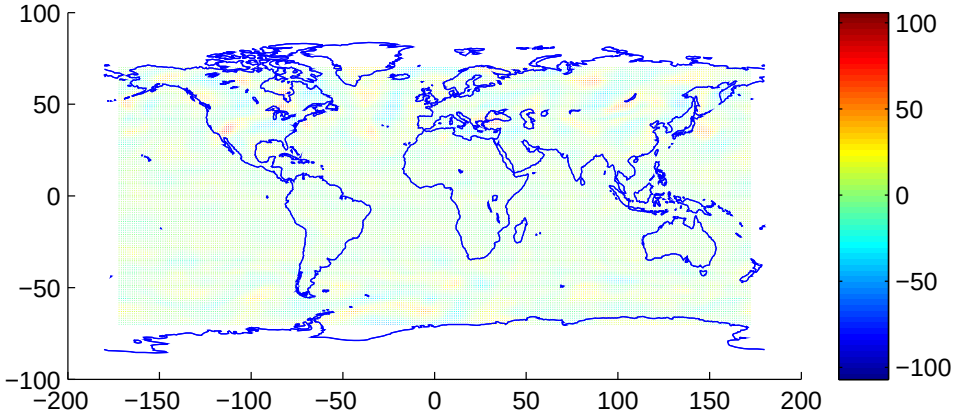


(a) Variance parameter σ^2 (b) Variance of error parameter τ^2

Figure A-11: The posterior distribution of the variance parameter in the 3D model

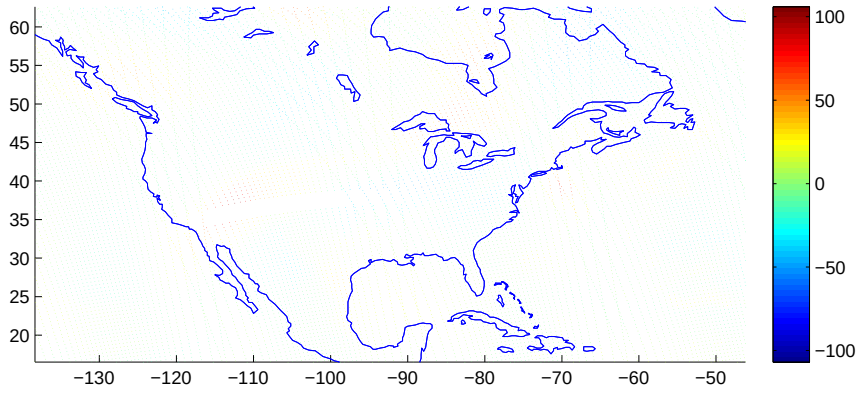


(a) Level 2

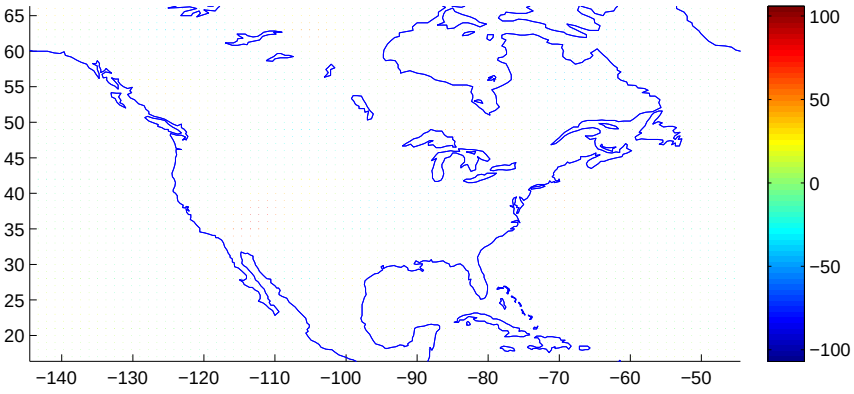


(b) Level 3

Figure A-12: Level 2 and Level 3 TOMS data



(a) Level 2 US



(b) Level 3 US

Figure A-13: Level 2 and Level 3 TOMS data for the US

VITA

Bledar Konomi was born in Albania. He majored in statistics at Athens University of Economics and Business, where he obtained his Bachelor of Science in 2005. He received his Ph.D. in statistics from Texas A&M University in December 2011. His research interests include Bayesian statistics, modeling spatial and spatial-temporal dataset, image analysis, dimension reduction, Markov-chain Monte-carlo computations.

He may be reached at:

Department of Statistics

Advisor: Dr. Bani Mallick

Texas A&M University

College Station, TX 77843-3143.

His web page URL is `http://stat.tamu.edu/~alexandros`

and his email address is `alex2433gr@yahoo.gr`.