STATISTICAL METHODS FOR LARGE SPATIAL AND SPATIO-TEMPORAL DATASETS

A Dissertation

by

BOHAI ZHANG

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

DOCTOR OF PHILOSOPHY

Chair of Committee,	Jianhua Huang
Co-Chair of Committee,	Huiyan Sang
Committee Members,	Mikyoung Jun
	Renyi Zhang
Head of Department,	Valen Johnson

August 2015

Major Subject: Statistics

ABSTRACT

Classical statistical models encounter the computational bottleneck for large spatial/spatio-temporal datasets. This dissertation contains three articles describing computationally efficient approximation methods for applying Gaussian process models to large spatial and spatio-temporal datasets.

The first article extends the FSA-Block approach in [60] in the sense of preserving more information of the residual covariance matrix. By using a block conditional likelihood approximation to the residual likelihood, the residual covariance of neighboring data blocks can be preserved, which relaxes the conditional independence assumption of the FSA-Block approach. We show that the approximated likelihood by the proposed method is Gaussian with an explicit form of covariance matrix, and the computational complexity is linear with sample size n. We also show that the proposed method can result in a valid Gaussian process so that both the parameter estimation and prediction are consistent in the same model framework. Since neighborhood information are incorporated in approximating the residual covariance function, simulation studies show that the proposed method can further alleviate the mismatch problems in predicting responses on block boundary locations.

The second article is the spatio-temporal extension of the FSA-Block approach, where we model the space-time responses as realizations from a Gaussian process model of spatio-temporal covariance functions. Since the knot number and locations are crucial to the model performance, a reversible jump Markov chain Monte Carlo (RJMCMC) algorithm is proposed to select knots automatically from a discrete set of spatio-temporal points for the proposed method. We show that the proposed knot selection algorithm can result in more robust prediction results. Then the proposed method is compared with weighted composite likelihood method through simulation studies and an ozone dataset.

The third article applies the nonseparable auto-covariance function to model the computer code outputs. It proposes a multi-output Gaussian process emulator with a nonseparable auto-covariance function to avoid limitations of using separable emulators. To facilitate the computation of nonseparable emulator, we introduce the FSA-Block approach to approximate the proposed model. Then we compare the proposed method with Gaussian process emulator with separable covariance models through simulated examples and a real computer code.

ACKNOWLEDGEMENTS

I would like to gratefully and sincerely thank my advisors Dr. Jianhua Huang and Dr. Huiyan Sang, for their guidance, supports, and encouragements during my Ph.D. study. Your passion, inspiration, generosity, and patience make me have a wonderful journey in statistics, and also let me determine to pursue the academic career. Without your help, I would never accomplish so much. It is my honor to work with both of you and I will cherish this experience.

I would also like to thank my committee members Dr. Mikyoung Jun and Dr. Renyi Zhang, for your helpful comments in my preliminary exam. Your valuable suggestions help me improve my dissertation and make it more complete.

I would like to thank the Department of Statistics at Texas A&M University, for its wonderful Ph.D. program, excellent faculty members, and powerful computing resources. I would also like to thank the SGSA, for organizing department events and hosting company recruitments.

Finally, I would like to thank my parents, for their understandings and supports during my graduate study. Also a big thank you to my deer friends; without your company and kind help, it would be much harder to complete a Ph.D. degree.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Gaussian Process Model

Spatial and spatio-temporal datasets are widely observed in many disciplines, such as the climatology, geology and so on, where the datasets are labeled by spatial coordinates such as longitude and latitude, as well as time (for space-time data). An important characteristic of spatial/spatio-temporal data is that nearby observations (in space or space-time) tend to be more alike than those far apart [15]. For example, one simple way to forecast tomorrow's weather is to use today or last few days' weather; similar conclusion holds for the spatial data, such as studies in environments (i.e., ground pollutants, distribution of species). To characterize the dependence of the responses, the covariance functions are widely used. The interests lie in inferences of the spatial/spatio-temporal dependence structures and subsequently making predictions on unobserved locations.

The responses are usually treated as realizations of an underlying spatial/spatio-temporal process, and the most popular process model is the Gaussian process model. Let $\mathscr{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a set of locations, where $\mathbf{x}_i = \mathbf{s}_i$ for spatial data and $\mathbf{x}_i = (\mathbf{s}_i, t_i)$ for space-time data. Let $\mathbb{Y} = (y(\mathbf{x}_1), y(\mathbf{x}_2), \ldots, y(\mathbf{x}_n))^T$ be a column vector collecting the responses at $\mathscr{X}$, then we assume that $\mathbb{Y} \sim \mathcal{N}(Z\boldsymbol{\beta}, \mathcal{C}(\boldsymbol{\theta}))$, where Z is the $n \times p$ design matrix, $\boldsymbol{\beta}$ is a $p \times 1$ regression coefficients vector, and $\mathcal{C}(\boldsymbol{\theta})$ is the covariance matrix depending on parameter vector $\boldsymbol{\theta}$. $\mathcal{C}(\boldsymbol{\theta})$ is usually assumed to be generated from a positive-definite covariance function, such as Matérn covariance model [51]:

$$\mathcal{C}(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) = \frac{\sigma^2}{\Gamma(\nu)} 2^{1-\nu} (h/\phi)^{\nu} K_{\nu}(h/\phi),$$

where $h = \|\mathbf{x}-\mathbf{x}'\|$; $\Gamma(\cdot)$ is the gamma function, $K_\nu(\cdot)$ is the modified Bessel function of the second kind, σ^2 is the variance parameter, $\phi > 0$ is the dependence range parameter, and $\nu > 0$ is the smoothness parameter. The log-likelihood function up to a constant is

$$\ell(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = -\frac{1}{2}|\mathcal{C}(\boldsymbol{\theta})| - \frac{1}{2}(\mathbb{Y} - Z\boldsymbol{\beta})^T\mathcal{C}^{-1}(\boldsymbol{\theta})(\mathbb{Y} - Z\boldsymbol{\beta}).$$

Given a predictive location $\mathbf{x}_p$ and assume all parameters are known, then we have the Best Linear Unbiased Predictor (BLUP),

$$\hat{y}(\mathbf{x}_p) = z^T(\mathbf{x}_p)\boldsymbol{\beta} + \mathcal{C}_{pn}\mathcal{C}^{-1}(\boldsymbol{\theta})(\mathbb{Y} - Z\boldsymbol{\beta})$$

and the corresponding Mean Squared Error (MSE)

$$\text{MSE}(\hat{y}(\mathbf{x}_p)) = \mathcal{C}_p - \mathcal{C}_{pn}\mathcal{C}^{-1}(\boldsymbol{\theta})\mathcal{C}_{pn}^T,$$

where $z(\mathbf{x}_p)$ is the $p \times 1$ vector of covariates at $\mathbf{x}_p$, $\mathcal{C}_p = \mathcal{C}(0; \boldsymbol{\theta})$ and $\mathcal{C}_{pn} = \mathcal{C}(\mathbf{x}_p, \mathscr{X}; \boldsymbol{\theta})$.

From the log-likelihood function and the BLUP formula, $\mathcal{C}^{-1}(\boldsymbol{\theta})$ and $|\mathcal{C}(\boldsymbol{\theta})|$ need to be computed for model inference and prediction. To compute the inverse and determinant of the covariance matrix, we need to obtain the Cholesky factor $L(\boldsymbol{\theta})$ such that $\mathcal{C}(\boldsymbol{\theta}) = L(\boldsymbol{\theta})L^T(\boldsymbol{\theta})$, which typically requires $\mathcal{O}(n^3)$ Floating-points Operations Per Second (flops). Since the computation costs grow quickly with the sample size, the direct application of Gaussian process models to large spatial/spatio-temporal models can be computationally prohibitive for large datasets. This computational challenge motives the innovations of new statistical methods scalable to handle large datasets [67].

1.2 Popular Approximation Methods for Gaussian Process Model

In this Section, we will give brief summaries of Gaussian process approximation methods most related to this dissertation. The introduction part in each chapter will give more comprehensive literature reviews of related methods.

1.2.1 Low-rank approximation models

The basic idea of low-rank models is to project the original process $y(\mathbf{x})$ to a low-dimensional space spanned by a small number of basis functions. Then the low-rank approximation methods seek to replace the original covariance matrix with an approximated low-rank matrix for computational efficiency. The popular low-rank models include the Predictive Process model [4, 21] and the Fixed Rank Kriging (FRK) model [13, 40]. Given a set of locations $\mathscr{X}^* = \{x_1^*, \ldots, x_m^*\}$, referred to as knots, the predictive process model approximates a zero-mean Gaussian process $y(\mathbf{x})$ using the conditional mean process $E(y(\mathbf{x})|y(\mathscr{X}^*))$, where $y(\mathscr{X}^*) = (y(\mathbf{x}_1^*), y(\mathbf{x}_2^*), \ldots, y(\mathbf{x}_m^*))^T$. The approximated covariance matrix by the predictive process model is

$$\mathcal{C}_{pp} = \mathcal{C}_{n*}\mathcal{C}_*^{-1}\mathcal{C}_{n*}^T,$$

where $\mathcal{C}_{n*} = \mathcal{C}(\mathscr{X}, \mathscr{X}^*; \boldsymbol{\theta})$ and $\mathcal{C}_* = \mathcal{C}(\mathscr{X}^*, \mathscr{X}^*; \boldsymbol{\theta})$. $\mathcal{C}_{pp}$ is positive semi-definite with rank m. When the responses are observed with white noises, fast computations can be achieved by Sherman-Woodbury-Morrison inversion formula [35] with computational complexity $\mathcal{O}(nm^2)$. Thus a small number of knots lead to significant reduction of computations.

Similarly to predictive process model, the FRK model assumes $y(\mathbf{x}) = S^T(\mathbf{x})\boldsymbol{\eta}$ based on a small number of basis functions, where $S(\mathbf{x}) = (S_1(\mathbf{x}), \ldots, S_r(\mathbf{x}))^T$. $\boldsymbol{\eta}$ is a $r \times 1$ random vector following a Gaussian distribution $\mathcal{N}(\mathbf{0}, K)$, where K is a $r \times r$ positive definite matrix to be estimated. The computational complexity of the FRK model is $\mathcal{O}(nr^2)$, so it has good computational scalability. Also by construction of the covariance matrix, the FRK model can handle large datasets with nonstationary dependence structures.

Although the low-rank approximations has computational complexity linear with sample size n, it has several limitations. For example, when the dependence range of the data is small, the low-rank model usually needs a large number of basis functions for providing a satisfactory approximation to the original process. [63] gives a comprehensive discussion of limitations of low-rank models and points out that the best low-rank model can perform poorly when the data are strongly correlated for neighboring observations.

1.2.2 Sparse approximation methods

The sparse approximation methods replace the correlations of distance locations by zeros and only keep the correlations among neighboring locations. Covariance tapering [23, 43] is a popular method to induce sparsity for the covariance matrix. Given a covariance function $\mathcal{C}(\cdot, \cdot; \boldsymbol{\theta})$, the tapered covariance is

$$\mathcal{C}_{taper}(\mathbf{x}, \mathbf{x}') = \mathcal{C}(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})\mathcal{K}(\|\mathbf{x} - \mathbf{x}'\|; \gamma),$$

where $\mathcal{K}(\mathbf{x}, \mathbf{x}'; \gamma)$ is the tapering function which is positive definite and has zero values when $\|\mathbf{x} - \mathbf{x}'\| \geq \gamma$. Thus a small γ leads to a sparse covariance matrix when evaluating $\mathcal{C}_{taper}(\mathbf{x}, \mathbf{x}')$ on observed set $\mathscr{X}$. The matrix decomposition algorithms for sparse matrices are applied subsequently to compute the Cholesky factor of the approximated covariance matrix by the tapered covariance function. The computational complexity is $\mathcal{O}(nr^2)$, where r is the number of nonzero entries per row and column of a sparse matrix. By construction, the covariance tapering approach ignores the dependence of responses far apart, thus its performance is poor when the dependence range is large.

1.2.3 Composite likelihood approach

The composite likelihood approach approximates the full data likelihood function by a product of low-dimensional marginal or conditional likelihoods [49, 70]. Since evaluations of low-dimensional likelihoods are computationally cheaper, the composite likelihood approach gains the computational efficiency. One simple composition likelihood approach is the independent blocks approximation [8, 63]. Given a partition of observed responses $\mathbb{Y} = \cup_{k=1}^{K} \mathbf{Y}_k$, the independence blocks approximation approximates the full likelihood $L(\mathbb{Y}; \boldsymbol{\theta})$ by $\prod\limits_{k=1}^{K} L(\mathbf{Y}_k; \boldsymbol{\theta})$. If we choose relatively equal block size n_b for each data block, the computational complexity of the independent blocks approximation is $\mathcal{O}(nn_b^2)$, so fast computations can be achieved if we choose n_b to be small.

Alternatively, [72, 64] approximates $L(\mathbb{Y}; \boldsymbol{\theta})$ by a product of conditional likelihoods of data blocks, referred to as the block conditional likelihood approximation. Motivated by the chain rule $L(\mathbb{Y}; \boldsymbol{\theta}) = \prod\limits_{k=1}^{K} L(\mathbf{Y}_k | \mathbf{Y}_{(k)}, \boldsymbol{\theta})$, where $\mathbf{Y}_{(k)} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_{k-1}\}$ for $k \geq 1$ and $\mathbf{Y}_{(1)} = \emptyset$, the block conditional likelihood approximation approximates $L(\mathbb{Y}; \boldsymbol{\theta})$ by

$$\prod_{k=1}^{K} L(\mathbf{Y}_k | \mathbf{Y}_{N(k)}, \boldsymbol{\theta}),$$

where $\mathbf{Y}_{N(k)} \subseteq \mathbf{Y}_{(k)}$ contains all the neighboring data blocks for kth data block. Computations of evaluating this approximated likelihood can be greatly reduced if $\mathbf{Y}_{N(k)}$ contains a small number of observations. Most recently, [18] proves that the conditional likelihood approximation approach in [72, 64] for $K = n$ can lead to a valid Gaussian likelihood. In addition, they show that a valid Gaussian process can be obtained by this approximation so that both parameter estimation and prediction of the proposed model can be performed in a unified framework.

1.3 Overall Structure

The following is the general structure of this dissertation. Section 2 extends the Full-Scale Approximation with Block modulating function [60], referred to as the FSA-Block approach, in the sense of preserving more information of the residual covariance function. Given a partition $\mathbb{Y} = \cup_{k=1}^{K} \mathbf{Y}_k$, the FSA-Block approach assumes that $\mathbf{Y}_k$'s are conditionally independent given the predictive process component. The conditional independence assumption may be strong when the predictive process part does not perform well (e.g, the number of knots is small or the dependence structure is local). By using the block conditional likelihood approximation to the full residual likelihood, we show that the residual covariance among neighboring $\mathbf{Y}_k$'s can be preserved. Since more information are kept for the residual covariance, the proposed method enjoys better statistical efficiency. We also show that the proposed method can result in a valid Gaussian process model so that the parameter estimation and prediction are consistent. We compare the proposed method with the FSA-Block approach and the block version of the nearest neighbor process approach [18] through simulation studies and a real precipitation dataset.

Section 3 is a spatio-temporal extension of the FSA-Block approach, where we consider modeling the space-time responses by a Gaussian process with a spatial-temporal covariance function. In addition, we discuss selection methods of the knot set for the proposed method and introduce a Reversible Jump Markov chain Monte Carlo algorithm [33] to dynamically update the knot number and knot locations. We demonstrate the effectiveness of proposed method using a simulated nonstationary dataset and an ozone data of eastern US.

Section 4 applies the FSA-Block approach to approximating the Gaussian process emulator for large computer code outputs. A multi-output Gaussian process emulator with a nonseparable auto-covariance function is proposed to avoid limitations of using separable emulators. To facilitate the computation of nonseparable emulator, the FSA-Block approach is applied to approximating the nonseparable auto-covariance function. We compare the performance of the proposed method with Gaussian process with separable auto-covariance function through simulated examples and a real computer code of the carbon capture system.

Summary and discussion of potential extensions are given in Section 5. The proofs of theorems in Section 2 and the algorithm of calculating the posterior distributions in Section 4 are provided in the Appendix.

2. A SMOOTH FULL-SCALE APPROXIMATION APPROACH FOR LARGE SPATIAL DATASETS

2.1 Introduction

Spatial datasets arising from ecology, climatology, and other disciplines have generated considerable interests for scientists. With the advent of remote sensing and GPS techniques, the spatial data collection capacity increases dramatically and statisticians nowadays are facing a large number of observations on variables of interest. The growth in data size imposes computational challenges to the classical statistical modeling methods [62, 5] and has driven the innovations of new methods scalable to handle large datasets [67].

One of the most popular models for spatial datasets is the Gaussian process model, assuming finite observations are jointly Gaussian. Although the Gaussian process model enjoys the mathematical tractability and can provide prediction intervals for observations on unobserved locations, its computational complexity generally grows cubically with the sample size n, due to the expensive matrix factorizations. Specifically, the calculations of inverse and determinant of the data covariance involve the Cholesky decomposition of the finite sample covariance matrix, whose computation requires $\mathcal{O}(n^3)$ floating point operations (flops). The evaluation of the Gaussian process model will be computationally prohibitive for very large n.

When the covariance matrix has a certain structure, such as the Toeplitz matrix [79], fast computations are available for evaluating the Gaussian process model. However, since the spatial process is generally observed at irregularly spaced locations and the dependence of distant pairs of observations is often nonnegligible, the data covariance matrix does not have any structures in general. Approaches tackling the computational challenge have adopted two major paths. The first path is to approximate the full likelihood function by some simplified versions. The composite likelihood approach [49, 70], as a general class of pseudo-likelihoods, has been used to model spatial datasets. The idea is to approximate the ordinary likelihood using products of marginal or conditional likelihoods of reduced dimensions. For marginal composite likelihood approach, [16] proposed a composite likelihood approach based on marginal densities of pairwise differences of responses; also based on bivariate marginal densities, [69] proposed a pairwise composite likelihood approach for spatial generalized linear mixed models. Recently, [19] proposed to use a composite likelihood function defined as the product of joint densities of pairwise spatial blocks and it enjoys better statistical efficiency than the composite likelihood based on bivariate marginal densities. For conditional composite likelihood approach, [72] and [64] constructed composite estimating functions based on conditional densities of spatial data blocks.

The second path is to approximate the data covariance matrix with either a low-rank or a sparse matrix whose matrix factorizations are computationally cheaper. The popular low-rank models include the Gaussian predictive process model [4] and the Fixed Rank Kriging model [13, 42], where the original spatial process is approximated by a smoother process based on a small number of basis functions. For sparse matrix approximations, the covariance tapering method [23, 43] approximates the original covariance with a sparse matrix by shrinking the dependence of distant pairs of spatial locations to be zero. Then the algorithms for manipulating sparse matrices are applied to reduce the computational burden. Instead of working on the covariance matrix, the Gaussian Markov Random Field model [56, 48] induces a sparse precision matrix for facilitating computations.

Since the low-rank models may fail to model the local variations well [21, 63] and the covariance tapering method, on the contrary, can not capture the large-scale dependence well, [59] proposed a so-called Full-Scale Approximation approach (F-SA), which adds a sparse residual covariance component to the covariance of the Gaussian predictive process model, for approximating the data covariance well under both large and small scale dependence scenarios. Specifically, let $\mathcal{C}(\cdot, \cdot; \boldsymbol{\theta})$ and $\mathcal{C}_l(\cdot, \cdot; \boldsymbol{\theta})$ be the covariance functions of the original Gaussian process and Gaussian predictive process, respectively, then the covariance function of the FSA approach is $\mathcal{C}^\dagger = \mathcal{C}_l(\cdot, \cdot; \boldsymbol{\theta}) + (\mathcal{C}(\cdot, \cdot; \boldsymbol{\theta}) - \mathcal{C}_l(\cdot, \cdot; \boldsymbol{\theta}))\mathcal{K}(\cdot, \cdot)$, where the function $\mathcal{K}$, referred to as the modulating function, is positive semi-definite and has a large number of zeros evaluated on observed spatial locations. If we choose $\mathcal{K}$ such that the residual covariance is block-diagonal, then the method is called the FSA-Block approach; if some compactly supported covariance functions are chosen for $\mathcal{K}$, then the method is called the FSA-Taper approach. [60] has shown that the FSA-Block approach can have better numerical results than the FSA-Taper approach.

This Section extends the FSA-Block approach in the sense of preserving more information of the residual covariance. By using a conditional likelihood approximation, the dependence across blocks of the residual covariance is preserved by the new proposed approach. Since more information are kept for the data covariance matrix, we expect that the new proposed approach can perform better than the FSA-Block approach when the residual dependence across blocks is not negligible. In addition, the proposed method can produce a smoother prediction surface than that by the FSA-Block method by using the conditional likelihood approximation, so the mismatches of predictions on boundary locations can be alleviated for the proposed method. We name the new proposed method the Smooth Full-Scale Approximation approach (SFSA). The approximated data covariance of the SFSA approach can reduce to covariance of the FSA-Block approach and the covariance of the conditional likelihood approximation approach proposed in [18]. Moreover, we show that the SFSA approach can define a valid Gaussian process, thus both the parameter estimation and prediction can be performed under the unified framework. Since the covariance function is available for the SFSA approach, the kriging formula can be directly applied for predictions.

2.2 Methodology

2.2.1 The spatial regression model

Let $Y(\mathbf{s})$ be a response variable observed at a location $\mathbf{s}$, where $\mathbf{s}$ belongs to the spatial domain $\mathcal{S} \subseteq \mathbb{R}^2$. We model $Y(\mathbf{s})$ through the following spatial regression model

$$Y(\mathbf{s}) = x^T(\mathbf{s})\boldsymbol{\beta} + w(\mathbf{s}) + \epsilon(\mathbf{s}), \tag{2.1}$$

where $x(\mathbf{s})$ is a $p \times 1$ vector of covariates, $\boldsymbol{\beta}$ is the corresponding regression coefficients vector, $w(\mathbf{s})$ is a latent zero-mean Gaussian process, and $\epsilon(\mathbf{s})$ is a Gaussian white noise process with constant variance τ^2, independent of $w(\mathbf{s})$. The covariance of the spatial process $w(\mathbf{s})$ characterizes the spatial dependence structure and it is specified by a valid covariance function $\mathcal{C}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) = \mathrm{Cov}(w(\mathbf{s}), w(\mathbf{s}'))$. For example, the Matérn covariance is widely used in spatial statistics due to its flexibility of modeling the smoothness of the spatial process,

$$\mathcal{C}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) = \frac{\sigma^2}{\Gamma(\nu)} 2^{1-\nu} (h/\phi)^{\nu} K_{\nu}(h/\phi), \tag{2.2}$$

where Γ is the gamma function, K_ν is the modified Bessel function of the second kind, σ^2 is the variance parameter, ϕ is the spatial dependence range parameter and ν is the smoothness parameter. The variance τ^2 of $\epsilon(\mathbf{s})$ is often referred to as the "nugget" effect, accounting for the measurement error effect.

Suppose $Y(\mathbf{s})$ is observed at n spatial locations $S = \{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$. Let $\mathbb{Y} = (Y(\mathbf{s}_1), \ldots, Y(\mathbf{s}_n))^T$ denote the $n \times 1$ observed response vector and $\mathbf{X} = (x(\mathbf{s}_1), \ldots, x(\mathbf{s}_n))^T$ denote the $n \times p$ design matrix. Then the log-likelihood function is

$$\ell(\mathbb{Y}|\boldsymbol{\theta}, \boldsymbol{\beta}) = -\frac{1}{2}|\mathcal{C}_{\mathbb{Y}}| - \frac{1}{2}(\mathbb{Y} - \mathbf{X}\boldsymbol{\beta})^T \mathcal{C}_{\mathbb{Y}}^{-1}(\mathbb{Y} - \mathbf{X}\boldsymbol{\beta}) + \text{constant}, \tag{2.3}$$

where the data covariance $\mathcal{C}_{\mathbb{Y}} = \mathcal{C}_w + \tau^2 I_n$ and $\mathcal{C}_w = [\mathcal{C}(\mathbf{s}_i, \mathbf{s}_j)]_{i,j=1,\ldots,n}$ is the covariance matrix of $w(\mathbf{s})$ on S. Evaluating (2.3) requires $\mathcal{O}(n^3)$ flops for calculating $|\mathcal{C}_{\mathbb{Y}}|$ and $\mathcal{C}_{\mathbb{Y}}^{-1}$ in general, so the computational cost can be very intensive or even prohibitive when n is large.

2.2.2 The FSA-Block approach

Given m locations $S^* = \{\mathbf{s}_1^*, \ldots, \mathbf{s}_m^*\}$, referred to as knots, the predictive process model [4] approximates $w(\mathbf{s})$ by the conditional mean process $w_l(\mathbf{s}) = E(w(\mathbf{s})|\mathbf{w}^*)$, where $\mathbf{w}^* = w(S^*) = (w(\mathbf{s}_1^*), \ldots, w(\mathbf{s}_m^*))^T$. Since $w(\mathbf{s})$ has zero mean, then $w_l(\mathbf{s}) = \mathcal{C}(\mathbf{s}, \mathcal{S}^*)\mathcal{C}_*^{-1}\mathbf{w}^*$, where $\mathcal{C}(\mathbf{s}, \mathcal{S}^*) = [\mathcal{C}(\mathbf{s}, \mathbf{s}_i^*)]_{i=1,\ldots,m}$ and $\mathcal{C}_* = [\mathcal{C}(\mathbf{s}_i^*, \mathbf{s}_j^*)]_{i,j=1,\ldots,m}$. The spatial process $w(\mathbf{s})$ can be decomposed into two independent processes

$$w(\mathbf{s}) = w_l(\mathbf{s}) + w_s(\mathbf{s}),$$

where $w_s(\mathbf{s})$ is the exact residual process of $w(\mathbf{s})$. Since the covariance function of $w_l(\mathbf{s})$ is $\mathcal{C}_l(\mathbf{s}, \mathbf{s}') = \mathcal{C}(\mathbf{s}, S^*)\mathcal{C}_*^{-1}\mathcal{C}^T(\mathbf{s}', S^*)$, the covariance function of $w_s(\mathbf{s})$ is $\mathcal{C}_s(\mathbf{s}, \mathbf{s}') = \mathcal{C}(\mathbf{s}, \mathbf{s}') - \mathcal{C}_l(\mathbf{s}, \mathbf{s}')$. By Schur complement property of linear algebra, $\mathcal{C}_s$ is positive definite when $S \cap S^* = \emptyset$ and positive semi-definite otherwise. Therefore, if we approximate $w(\mathbf{s})$ only using $w_l(\mathbf{s})$, the covariance information in $w_s(\mathbf{s})$ will be lost. The lost of information of spatial dependence can be severe when the fine scale variations of the process are not negligible [21, 63].

Let $\mathcal{C}_{w_l} = \mathcal{C}(S, S^*)\mathcal{C}_*^{-1}\mathcal{C}^T(S, S^*)$ denote the covariance matrix of the predictive process $w_l(\mathbf{s})$ on S, then the exact residual covariance matrix is $\mathcal{C}_{w_s} = \mathcal{C}_w - \mathcal{C}_{w_l}$. This residual covariance is the covariance matrix of the conditional density $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta} + \mathcal{C}(S, S^*)\mathcal{C}_*^{-1}\mathbf{w}^*, \mathcal{C}_{w_s} + \tau^2 I)$, up to a matrix proportional to an identity matrix. Since $\mathcal{C}_{w_s}$ in general is a dense matrix, evaluating this conditional density will be computationally expensive for large n. Since $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = \int p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^*$, if we substitute some valid Gaussian density whose computational complexity is cheaper than $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$, then after integrating out $\mathbf{w}^*$, an approximated Gaussian data likelihood can be readily obtained. Compared with the covariance matrix of $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta})$, the covariance matrix of $\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*$ is closer to a sparse matrix, Since it has smaller off-diagonal entries due to subtracting $\mathcal{C}_{w_l}$ from $\mathcal{C}_w + \tau^2 I$. This observation leads to the independent blocks approximation to $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$.

Specifically, given a partition rule $\mathcal{P}$ leading to a partition of locations $S = \cup_{k=1}^K S_k$, let the corresponding partition of observations be $\mathbb{Y} = \cup_{i=1}^K \mathbf{Y}_i$ and $\mathbf{Y}_i s$ have relatively equal size n_i. We assume the data blocks $\mathbf{Y}_i$ are independent given $\mathbf{w}^*$ and model parameters, that is $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) = \prod_{k=1}^K p(\mathbf{Y}_k|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$. The FSA-Block approximation to the data likelihood function is

$$p_{FSAB}(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = \int_{\mathbf{w}^*} \prod_{i=1}^{K} p(\mathbf{Y}_k|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^*.$$

If we group observations properly, then $p_{FSAB}(\mathbb{Y}|\boldsymbol{\theta}, \boldsymbol{\beta}) \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, (\mathcal{C}_{w_l} + \mathcal{C}_{w_s} \circ \mathcal{T}_B + \tau^2 I))$, where $\mathcal{T}_B$ is a block-diagonal matrix with $\mathbf{1}_{n_i}\mathbf{1}_{n_i}^T$ as its i^{th} block, and $\circ$ is the Schur product of two matrices. Compared with the approximated covariance $\mathcal{C}_{w_l}$ of the predictive process model, an additional block-diagonal residual covariance is added to correct the approximation errors within data blocks. Since $\mathcal{C}_{w_s} \circ \mathcal{T}_B$

plus $\tau^2 I$ is still block-diagonal, it takes $\mathcal{O}(n)$ order flops to compute its inverse and determinant. By using the Sherman-Woodbury-Morrison inversion formula, it can be shown that the computational complexity of the FSA-Block approach is linear with n [59].

2.2.3 The proposed approach

The independent blocks approximation to $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$ ignores the residual dependence across blocks. The loss of information can be severe when $w_l(\mathbf{s})$ does not approximate $w(\mathbf{s})$ well and the entries across blocks of the residual covariance are not negligible. In this case, preserving the large entries of the residual covariance across data blocks may gain better statistical efficiency. Motivated by the block conditional likelihood approach [64], we propose to use the block conditional likelihood approximation for $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$. Let $(k-1) = \{1, 2, \ldots, k-1\}$ and $\mathbf{Y}_{(k-1)} = (\mathbf{Y}_1^T, \ldots, \mathbf{Y}_{k-1}^T)^T$, then by chain rule

$$p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) = p(\mathbf{Y}_1|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \cdot \prod_{k=2}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{(k-1)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*).$$

When the sample size n is large, it will be computationally prohibitive to compute the full conditional density $p(\mathbf{Y}_k|\mathbf{Y}_{(k-1)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$ for large k. So we may let the conditioned set of block k be a subvector of $\mathbf{Y}_{(k-1)}$ [64],

$$\tilde{p}(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) = \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*), \tag{2.4}$$

where $\mathbf{Y}_{N(k)}$ is a $n_{N(k)} \times 1$ subvector of $\mathbf{Y}_{(k-1)}$ with location set $S_{N(k)}$, i.e., the neighboring observations of $\mathbf{Y}_k$ in $\mathbf{Y}_{(k-1)}$; let $S_{N(1)} = \emptyset$. In this paper, we consider the special case that $S_{N(k)}$ contains all locations in q nearest neighboring blocks of block k in terms of Euclidean distances of block centers. Specifically,

$$S_{N(k)} = \begin{cases} \emptyset, & \text{if } k = 1 \\ \{S_1, S_2, \ldots, S_{k-1}\}, & \text{if } k \leq q \\ q \text{ nearest neighboring blocks in} \{S_1, S_2, \ldots, S_{k-1}\}, & \text{if } k > q \end{cases}$$

If we choose $q \ll K$, then by using a conditional set of a reduced dimension, the computational cost of evaluating $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$ can be greatly reduced. The conditional likelihood approximation includes the independent blocks approximation as a special case, since the latter uses $\emptyset$ as the conditioned set for every $\mathbf{Y}_k$.

Let $U_k = \mathcal{C}(S_k, S^*)\mathcal{C}_*^{-1}$, $U_{N(k)} = \mathcal{C}(S_{N(k)}, S^*)\mathcal{C}_*^{-1}$, Σ_k denote the residual covariance of $w_s(S_k) + \epsilon(S_k)$, and $\Sigma_{N(k)}$ denote the covariance of $w_s(S_{N(k)}) + \epsilon(S_{N(k)})$. Then $p(\mathbf{Y}_k|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \sim \mathcal{N}(U_k\mathbf{w}^*, \Sigma_k)$ and $p(\mathbf{Y}_{N(k)}|\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \sim \mathcal{N}(U_{N(k)}\mathbf{w}^*, \Sigma_{N(k)})$. Let $\Sigma_{k,N(k)}$ denote the residual cross-covariance between $w_s(S_k)$ and $w_s(S_{N(k)})$, then by conditional normal facts,

$$\begin{aligned} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \quad \propto \quad & |\Sigma_{con}^{(k)}|^{-\frac{1}{2}} \exp(-\frac{1}{2}(\mathbf{Y}_k - U_k\mathbf{w}^* - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(\mathbf{Y}_{N(k)} - U_{N(k)}\mathbf{w}^*))^T \\ & \times\Sigma_{con}^{(k)^{-1}}(\mathbf{Y}_k - U_k\mathbf{w}^* - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(\mathbf{Y}_{N(k)} - U_{N(k)}\mathbf{w}^*))), \end{aligned}$$

where $\Sigma_{con}^{(k)} = \Sigma_k - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}\Sigma_{k,N(k)}^T$. Let

$$B_{kl} = \begin{cases} I_{n_k}, & \text{if } l = k; \\ -\Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(, n_{(l-1)} + 1 : n_{(l)}), & \text{if } l \in N(k); \\ \mathbf{0}, & \text{otherwise,} \end{cases} \tag{2.5}$$

where $n_{(l)} = \sum_{1 \leq i \leq l, i \in N(k)} n_i$. Let $B_k^* = (B_{k1}, \ldots, B_{kK})$, then $\mathbf{Y}_k - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}\mathbf{Y}_{N(k)} =$

$B_k^* \mathbb{Y}$ and $U_k - \Sigma_{k,N(k)} \Sigma_{N(k)}^{-1} U_{N(k)} = B_k^* U$. Therefore, the conditional density

$$p(\mathbf{Y}_k | \mathbf{Y}_{N(k)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \propto |\Sigma_{con}^{(k)}|^{-\frac{1}{2}} (\mathbb{Y} - U\mathbf{w}^*) B_k^{*^T} \Sigma_{con}^{(k)^{-1}} B_k^* (\mathbb{Y} - U\mathbf{w}^*).$$

The proposed method yields $\tilde{p}(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = \int_{\mathbf{w}^*} \prod_{k=1}^{K} p(\mathbf{Y}_k | \mathbf{Y}_{N(k)}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*) \cdot p(\mathbf{w}^* | \boldsymbol{\theta}) d\mathbf{w}^*$. The following Theorem 2.2.1 shows that this approximated likelihood is Gaussian with a closed-form covariance matrix.

Theorem 2.2.1. *If* $\mathbb{Y} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \mathcal{C}_{\mathbb{Y}})$, *then* $\tilde{p}(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \mathcal{C}_{\mathbb{Y}}^{\dagger})$, *where* $\mathcal{C}_{\mathbb{Y}}^{\dagger} = B^{-1} \Sigma_{con} B^{T^{-1}} + \mathcal{C}(S, S^*) \mathcal{C}_*^{-1} \mathcal{C}^T(S, S^*)$, *$B$ is lower-triangular, and Σ_{con} is block-diagonal.*

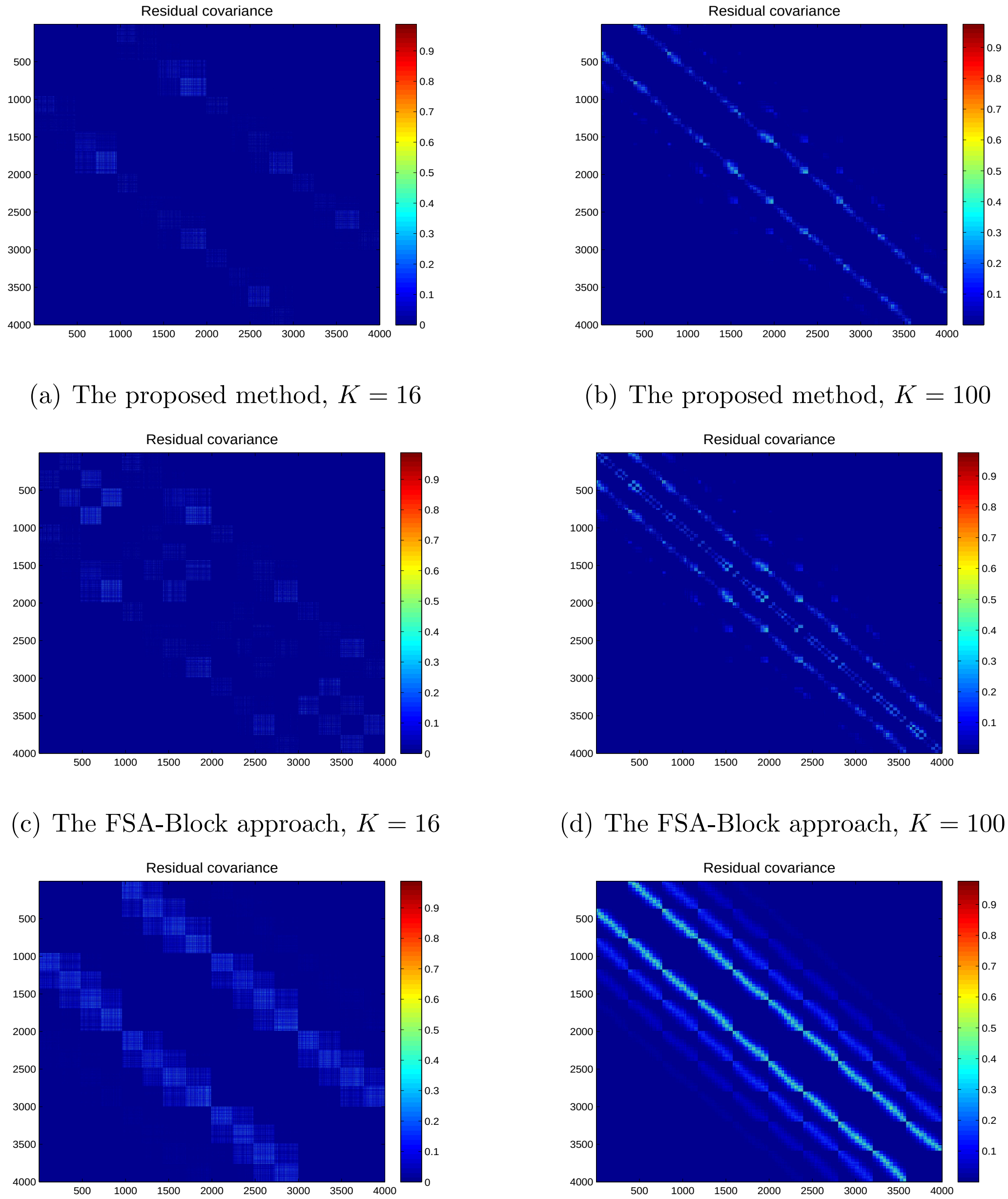
The proof is given in the Appendix. Σ_{con} is a block-diagonal matrix with $\Sigma_{con}^{(k)}$ as its k^{th} block and $B = (B_1^{*^T}, \ldots, B_K^{*^T})^T$. Since Σ_{con} is obtained based on the residual covariance function $\mathcal{C}_s(\mathbf{s}, \mathbf{s}') + \tau^2 \delta(\mathbf{s}, \mathbf{s}')$, where $\delta(\cdot, \cdot)$ is the Kronecker delta function, it is positive definite with rank n. Since $\mathcal{C}(S, S^*) \mathcal{C}_*^{-1} \mathcal{C}^T(S, S^*)$ is a positive semi-definite matrix by the predictive process model, the approximated data covariance $\mathcal{C}_{\mathbb{Y}}^{\dagger}$ is positive definite.

2.2.4 Relations to previous methods

The proposed method has connections with the FSA-Block approach and the block conditional approach by [64]. If we ignore the residual dependence across data blocks and assume $p(\mathbf{Y}_k | \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{w}^*)$ are independent for $k = 1, \ldots, K$, then the matrix B is an identity matrix and Σ_{con} is a block-diagonal matrix with k^{th} diagonal block $\Sigma_k = \mathcal{C}_s(S_k, S_k) + \tau^2 I_{n_k}$. In this case, the approximated data covariance by the new proposed method reduces to the covariance of the FSA-Block approach.

Following the proof in [18], the approximated data covariance by the block conditional likelihood approximation approach is $B^{-1} \Sigma_{con} B^{T^{-1}}$, where B and Σ_{con} have the same form as that in the proposed method but are calculated based on the original data covariance function $\mathcal{C}(\cdot,\cdot)+\tau^2\delta(\cdot,\cdot)$, instead of the residual covariance function $\mathcal{C}_s(\cdot,\cdot)+\tau^2\delta(\cdot,\cdot)$. It does not have the predictive process covariance $\mathcal{C}(S,S^*)\mathcal{C}_*^{-1}\mathcal{C}^T(S,S^*)$, because it assumes $\mathbf{Y}_k$ are independent given its neighboring observations, instead of assuming this is true conditional on $\mathbf{w}^*$.

Compared with the approximated data covariance by the FSA-Block approach, the proposed method can correct part of the approximation errors across data blocks; compared with the approximated data covariance by the block conditional likelihood approach, the proposed method does not totally ignore the dependence among non-neighboring blocks and uses the covariance of predictive process model as approximations. Therefore, the proposed method can induce a data covariance with smaller approximation errors. Figure 2.1 shows the absolute values of residual covariance matrix by three approaches, where the residual covariance is $\mathcal{C}_{\mathbb{Y}}-\tilde{\mathcal{C}}_{\mathbb{Y}}$ for some approximated data covariance matrix $\tilde{\mathcal{C}}_{\mathbb{Y}}$. Specifically, 4000 locations are randomly selected in a square domain $[0,10]\times[0,10]$ and the exponential model $\mathcal{C}(\mathbf{s},\mathbf{s}')=\exp(-\|\mathbf{s}-\mathbf{s}'\|)$ with nugget effect 0.01 is evaluated on these locations. The grid is used to create blocks and block numbers are in an increasing order from northwest to southeast. The locations within the same block are grouped together and the neighboring set is S_{k-1} for S_k. Compared with the FSA-Block approach, since the residual covariance between each block and its neighbors are corrected for the proposed method, we can observe that the entries of residual covariance matrix within a certain band are much smaller than those by the FSA-Block approach; compared with the conditional likelihood approach, while both methods provide good approximations for the covariance within blocks and between each block and its neighboring blocks, the proposed method leads to smaller entries of the residual covariance across blocks that are not neighbors due to including the covariance of the predictive process model.



(a) The proposed method, $K = 16$

(b) The proposed method, $K = 100$

(c) The FSA-Block approach, $K = 16$

(d) The FSA-Block approach, $K = 100$

(e) The CCL approach, $K = 16$

(f) The CCL approach, $K = 100$

Figure 2.1: Plots of residual covariance for 3 methods.

2.2.5 Computational complexity of the SFSA approach

Suppose all data blocks have an equal block size n_b, and each data block at most has q neighbors. From the Appendix, evaluating the likelihood of the proposed approach needs to compute the quadratic term $\mathbb{Y}^T B^T(\Sigma_{con}^{-1} - \Sigma_{con}^{-1} BU\Sigma_{\mathbf{w}^*}U^T B^T \Sigma_{con}^{-1})B\mathbb{Y}$, where $\Sigma_{\mathbf{w}^*} = (U^T B^T \Sigma_{con}^{-1} BU + \mathcal{C}_*^{-1})^{-1}$. The computation bottlenecks lie in computations of Σ_{con}^{-1}, BU, and $U^T B^T \Sigma_{con}^{-1} BU$. Since Σ_{con} is block-diagonal, its inverse takes $\mathcal{O}(Kn_b^3) = \mathcal{O}(nn_b^2)$ flops. BU has computational complexity $\mathcal{O}(nmqn_b)$, because B is a lower-triangular matrix with at most qn_b nonzero entries per row and U is a n by m matrix. The computation of $U^T B^T \Sigma_{con}^{-1} BU$ involves a product of a $m \times n$ matrix and a $n \times m$ matrix, so its computation complexity is $\mathcal{O}(nm^2)$. Therefore, the computational complexity of the proposed method has the order $\mathcal{O}(nn_b^2 + nmqn_b + nm^2)$. If we set the knot size $m \ll n$, the block size $n_b \ll n$, and $q \ll K$, then the proposed method has computational complexity linear with n.

2.2.6 Choices of tuning parameters

The FSA-Block approach needs to specify the knot set and a block partition; compared with the FSA-Block approach, the proposed SFSA approach has additional tuning parameters: ordering of blocks and number of neighboring blocks q. For knots selection, a heuristic way is to predetermine the knot number according to the balance of available computing resources and pilot studies of statistical efficiency, then to place the knots with a good space coverage. For example, we can use random sampling, Latin Hypercube Sampling [52] or a spatial grid for placing the knots. Alternatively, we can treat the knots as model parameters and select them adaptively [34, 41, 78]. For the block partition, the goal is to maximize block numbers while minimizing the residual correlations across blocks. [19] provides some guidance on blocking strategy, and one recommendation is to use the empirical variogram to determine the block width. If the residual covariance is fairly isotropic, the partition algorithm based on Euclidean distances of locations such as K-means clustering is a simple choice. Unfortunately, since the predictive process covariance is not isotropic, the residual covariance function is not isotropic too. In this case, applying clustering algorithm using the estimated residual covariance may be a better choice for block partition, but we stick to K-means clustering algorithm in this paper for its simplicity.

The ordering of blocks can also affect the performance of the SFSA approach. Since the neighbors of one block can only be the past blocks (blocks with a smaller block number), the choices of neighbors are more restricted for blocks of a relatively small block number. So it may gain better statistical efficiency if we guarantee that a block of a small block number has some really close past blocks (the closeness of two blocks is measured by the distance of block centers). one heuristic way for ordering blocks is first to number a block with the minimum distance to all other blocks S_1, then number its nearest neighboring block S_2; given a current set of numbered blocks $S^{(k)} = \{S_1, S_2, \ldots, S_k\}$, the block in the remaining blocks with the minimum distance to the set $S^{(k)}$ is numbered S_{k+1} and let $S^{(k+1)} = \{S^{(k)}, S_{k+1}\}$; we keep ordering the blocks until $k = K$. When the spatial locations are irregularly spaced such as the real dataset in Section 2.5, this heuristic ordering approach empirically works well. In Section 2.4, we illustrate the effect of number of neighboring blocks q. Based on the simulation results, a small number of neighboring blocks such as $q = 4$ (with several hundred neighboring observations) usually leads to performance very close to the full covariance model in terms of parameter estimation.

2.3 Parameter Estimation and Prediction

2.3.1 Maximum likelihood estimators

The maximum likelihood estimates of model parameters maximize the log-likelihood function (2.3). To facilitate computations, we replace the full covariance $\mathcal{C}_{\mathbb{Y}}$ with the approximated covariance $\mathcal{C}_{\mathbb{Y}}^{\dagger}$ in Theorem 2.2.1,

$$\ell(\mathbb{Y}|\boldsymbol{\theta}, \boldsymbol{\beta}) = -\frac{1}{2}|\mathcal{C}_{\mathbb{Y}}^{\dagger}| - \frac{1}{2}(\mathbb{Y} - \mathbf{X}\boldsymbol{\beta})^T \mathcal{C}_{\mathbb{Y}}^{\dagger^{-1}} (\mathbb{Y} - \mathbf{X}\boldsymbol{\beta}) + \text{constant}.$$

By the proof in Appendix,

$$\mathcal{C}_{\mathbb{Y}}^{\dagger^{-1}} = \Sigma_{con}^{-1} - \Sigma_{con}^{-1} B U \Sigma_{\mathbf{w}^*} U^T B^T \Sigma_{con}^{-1},$$

where $\Sigma_{\mathbf{w}^*} = (U^T B^T \Sigma_{con}^{-1} B U + \mathcal{C}_*^{-1})^{-1}$ is a $m \times m$ matrix, Σ_{con} is a block-diagonal matrix and B is a sparse lower-triangular matrix. So the computations $\mathcal{C}_{\mathbb{Y}}^{\dagger^{-1}}$ can be greatly reduced when we choose the knot size m, the block size n_B, and number of neighboring blocks q to be small. For the determinant,

$$|\mathcal{C}_{\mathbb{Y}}^{\dagger}| = |U^T B^T \Sigma_{con}^{-1} B U + \mathcal{C}_*^{-1}| \cdot |\Sigma_{con}| \cdot |\mathcal{C}_*|.$$

Efficient computations can be achieved since we only need to compute the determinants of two $m \times m$ matrices and a block-diagonal matrix.

2.3.2 Bayesian inference on model parameters

The Bayesian inference starts from the specifications of prior distributions of $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. The conjugate normal prior $\pi(\beta) \sim \mathcal{N}(\mu_0, \Sigma_0)$ can be assigned to $\boldsymbol{\beta}$. The priors of $\boldsymbol{\theta}$ depends on the form of the covariance function. Take the Matérn covariance model (2.2) as an example, the inverse gamma prior $IG(a, b)$ can be assigned to variance parameter σ^2 and the nugget τ^2 where hyper-parameters a, b are chosen with guesses of the mean and variance; for the dependence range parameter, a uniform prior with a reasonable support of practical dependence ranges can be used; for smoothness parameter ν, usually an uniform prior at $(0, 2]$ is used since high-order smoothness can hardly be identified for the real datasets.

We draw posterior samples of model parameters from the posterior $p(\boldsymbol{\beta}, \boldsymbol{\theta}|\mathbb{Y}) \propto p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta})\pi(\boldsymbol{\beta})\pi(\boldsymbol{\theta})$. The full conditional distribution of $\boldsymbol{\beta}$ has a closed-form $p(\boldsymbol{\beta}|\mathbb{Y}, \boldsymbol{\theta}) \sim \mathcal{N}(\mu_{\boldsymbol{\beta}|\cdot}, \Sigma_{\boldsymbol{\beta}|\cdot})$, where

$$\begin{aligned}\mu_{\boldsymbol{\beta}|\cdot} &= \Sigma_{\boldsymbol{\beta}|\cdot}(\mathbf{X}^T\mathcal{C}_{\mathbb{Y}}^{-1}\mathbb{Y} + \Sigma_0^{-1}\mu_0),\\ \Sigma_{\boldsymbol{\beta}|\cdot} &= (\mathbf{X}^T\mathcal{C}_{\mathbb{Y}}^{-1}\mathbf{X} + \Sigma_0^{-1})^{-1}.\end{aligned}$$

The Gibbs sampler is used to draw posterior samples from $\mathcal{N}(\mu_{\boldsymbol{\beta}|\cdot}, \Sigma_{\boldsymbol{\beta}|\cdot})$. Since the full conditional distributions of $\boldsymbol{\theta}$ don't have closed-forms, we use Metropolis-Hastings algorithm to draw posterior samples of $\boldsymbol{\theta}$. For very large sample size n, we replace $\mathcal{C}_{\mathbb{Y}}$ with the approximated covariance $\mathcal{C}_{\mathbb{Y}}^{\dagger}$ by the SFSA approach in $\mu_{\boldsymbol{\beta}|\cdot}, \Sigma_{\boldsymbol{\beta}|\cdot}$, and $p(\mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta})$, for obtaining posterior samples of model parameters.

2.3.3 Prediction

Let $S_p = \{\mathbf{s}_1, \ldots, \mathbf{s}_{n_p}\}$ be a set of predictive spatial locations such that $S_p \cap S = \emptyset$ and $\mathbf{Y}_p = (Y(\mathbf{s}_1), \ldots, Y(\mathbf{s}_{n_p}))^T$ be the corresponding vector of responses. Given the partition rule $\mathcal{P}$ that partitions S into K blocks, suppose it partitions S_p into $r \leq K$ distinct blocks $S_{p,k}$ with the block number M_k, $k = 1, \ldots, r$. We start from the joint density $p(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) = \int p(\mathbf{Y}_p|\mathbb{Y}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot p(\mathbb{Y}|\mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^*$. Let $\mathbf{Y}_{p,k}$ be the response vector at $S_{p,k}$, we define

$$\begin{aligned}\tilde{p}(\mathbf{Y}_p|\mathbb{Y}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) &= \prod_{k=1}^{r} \tilde{p}(\mathbf{Y}_{p,k}|\mathbb{Y}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta})\\ &= \prod_{k=1}^{r} p(\mathbf{Y}_{p,k}|\mathbf{Y}_{M_k}, \mathbf{Y}_{N(M_k)}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}). \qquad (2.6)\end{aligned}$$

This definition assumes $\mathbf{Y}_{p,k}$'s are conditional independent given $\mathbf{w}^*$ and their neighbors in $\mathbb{Y}$. By (2.6), the approximated joint density

$$\begin{aligned}\tilde{p}(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta}) &= \int \tilde{p}(\mathbf{Y}_p|\mathbb{Y}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot \tilde{p}(\mathbb{Y}|\mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^* \\ &= \int \prod_{k=1}^{r} p(\mathbf{Y}_{p,k}|\mathbf{Y}_{M_k}, \mathbf{Y}_{N(M_k)}, \mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot \tilde{p}(\mathbb{Y}|\mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^*,\end{aligned}$$

where $\tilde{p}(\mathbb{Y}|\mathbf{w}^*, \boldsymbol{\beta}, \boldsymbol{\theta})$ is the Gaussian density in (2.4). Then the approximated conditional density of $\mathbf{Y}_p|\mathbb{Y}, \boldsymbol{\beta}, \boldsymbol{\theta}$ can be directly derived from $\tilde{p}(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\beta}, \boldsymbol{\theta})$.

Theorem 2.3.1. *Let $\mathbf{X}_{p_k}$ denote the design matrix of $\boldsymbol{Y}_{p,k}$, U_{p_k} denote $\mathcal{C}(S_{p,k}, S^*)\mathcal{C}_*^{-1}$ and $\Sigma_{con}^{(p_k)}$ be the residual conditional variance of $\boldsymbol{Y}_{p,k}$, given its neighbors $\boldsymbol{Y}_{N(M_k)}$ and $\boldsymbol{Y}_{M_k}$. Define $B_{p_k} = (B_{p_k,1}, \ldots, B_{p_k,K})$, where $B_{p_k,l}$, $l = 1, \ldots, K$ has the same definition as (2.5). Let $\boldsymbol{X}_p = (\mathbf{X}_{p_1}^T, \ldots, \boldsymbol{X}_{p_r}^T)^T$, $U_p = (U_{p_1}^T, \ldots, U_{p_r}^T)^T$, $B_p = (B_{p_1}, \ldots, B_{p_r})$ and $\Sigma_{con}^p = \mathrm{diag}\{\Sigma_{con}^{(p_1)}, \ldots, \Sigma_{con}^{(p_r)}\}$, then the approximated conditional density*

$$\boldsymbol{Y}_p|\mathbb{Y}, \boldsymbol{\beta}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}_p, \Sigma_p),$$

where

$$\begin{aligned}\boldsymbol{\mu}_p &= \mathbf{X}_p\boldsymbol{\beta} + F_p\mathcal{C}_{\mathbb{Y}}^{\dagger^{-1}}(\mathbb{Y} - \mathbf{X}\boldsymbol{\beta}), \\ \Sigma_p &= \Sigma_{con}^p + B_pB^{-1}\Sigma_{con}B^{T^{-1}}B_p^T + U_p\mathcal{C}^*U_p^T - F_p\mathcal{C}_{\mathbb{Y}}^{\dagger^{-1}}F_p^T, \\ F_p &= (-B_pB^{-1}\Sigma_{con}B^{T^{-1}} + U_p\mathcal{C}^*U^T).\end{aligned}$$

The conditional mean $\boldsymbol{\mu}_p$ is the kriging formula for spatial predictions under the SFSA approximations. We postpone the proof of Theorem 2.3.1 to the next subsection 2.3.4 where we prove that the SFSA approach can induce a valid Gaussian process with a closed-form covariance function.

2.3.4 The Smooth FSA spatial process

Actually the SFSA approach can yield a valid spatial process so that both the parameter estimation and prediction can be performed in the same framework. In Section 2.2.2, we show that the underlying spatial process $w(\mathbf{s})$ can be decomposed into two independent processes $w_l(\mathbf{s})$ and $w_s(\mathbf{s})$, where $w_l(\mathbf{s})$ is the predictive process with covariance function $\mathcal{C}_l(\cdot,\cdot)$ and $w_s(\mathbf{s})$ is the exact residual process with covariance function $\mathcal{C}(\cdot,\cdot) - \mathcal{C}_l(\cdot,\cdot)$. Let $\tilde{w}_s(\mathbf{s}) = w_s(\mathbf{s}) + \epsilon(\mathbf{s})$ be the new residual process incorporating the nugget effect, then the data process $Y(\mathbf{s}) = x^T(\mathbf{s})\boldsymbol{\beta} + w_l(\mathbf{s}) + \tilde{w}_s(\mathbf{s})$. In the following we show that the proposed method approximates the exact residual process $\tilde{w}_s(\mathbf{s})$ by a block version of the nearest neighboring process [18]. Given a partition rule $\mathcal{P}$ leading to $S = \cup_{k=1}^{K} S_k$, the key assumption $\tilde{p}(\mathbb{Y}|\boldsymbol{\beta},\boldsymbol{\theta},\mathbf{w}^*) = \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)},\boldsymbol{\beta},\boldsymbol{\theta},\mathbf{w}^*)$ of the proposed approach is equivalent to $\tilde{p}(\tilde{w}_s(S)|\boldsymbol{\theta}) = \prod_{k=1}^{K} p(\tilde{w}_s(S_k)|\tilde{w}_s(S_{N(k)}),\boldsymbol{\theta})$, since $\tilde{w}_s(\mathbf{s})$ is the only random component given $\mathbf{w}^*$ and all model parameters. Thus for the SFSA approach, a block version of the nearest neighboring approximation is applied to the residual likelihood $p(\tilde{w}_s(S)|\boldsymbol{\theta})$. Following the results in [18], the resulting approximated residual covariance matrix is $B^{-1}\Sigma_{con}B^{T^{-1}}$, denoted by $\Sigma_{\mathbb{Y}}^{\dagger}$.

Let $\mathcal{P}$ partitions a set of predictive locations S_p into r distinct blocks $S_{p,k}$ and denote the block number of $S_{p,k}$ by M_k, $k = 1,\ldots,r$. We assume that

$$\tilde{p}(\tilde{w}_s(S_p)|\tilde{w}_s(S),\boldsymbol{\theta}) = \prod_{k=1}^{r} p(\tilde{w}_s(S_{p,k})|\tilde{w}_s(S_{M_k}),\tilde{w}_s(S_{N(M_k)}),\boldsymbol{\theta}).$$

This assumption is equivalent to $\tilde{p}(\mathbf{Y}_p|\mathbb{Y},\mathbf{w}^*,\boldsymbol{\theta}) = \prod_{k=1}^{r} p(\mathbf{Y}_{p,k}|\mathbf{Y}_{M_k},\mathbf{Y}_{N(M_k)},\mathbf{w}^*,\boldsymbol{\theta})$ in subsection 2.3.3. Then let $S_v \subset \mathcal{S}$ be any spatial location set and $S_u = S_v \setminus S$ be the predictive location set in S_v. Denote $\tilde{w}_s(S_v)$ by $\tilde{\mathbf{w}}_v$ and $\tilde{w}_s(S_u)$ by $\tilde{\mathbf{w}}_u$, we define

$$\tilde{p}(\tilde{\mathbf{w}}_v|\boldsymbol{\theta}) \;=\; \int \tilde{p}(\tilde{\mathbf{w}}_u|\tilde{w}_s(S),\boldsymbol{\theta})\tilde{p}(\tilde{w}_s(S)|\boldsymbol{\theta}) \prod_{\mathbf{s}_i\in S\setminus S_v} d\tilde{w}_s(\mathbf{s}_i). \tag{2.7}$$

It is easy to verify that the defined residual process, denoted by $\tilde{w}_s^{\dagger}(\mathbf{s})$, satisfies the conditions of Kolmogorov consistency theorem and is a valid spatial process [18]. According to the law of total covariance

$$\mathrm{Cov}(\tilde{w}_s^{\dagger}(\mathbf{s}),\tilde{w}_s^{\dagger}(\mathbf{s}')) = \mathrm{E}(\mathrm{Cov}(\tilde{w}_s^{\dagger}(\mathbf{s}),\tilde{w}_s^{\dagger}(\mathbf{s}')|\tilde{w}_s(S)))+\mathrm{Cov}(\mathrm{E}(\tilde{w}_s^{\dagger}(\mathbf{s})|\tilde{w}_s(S)),\mathrm{E}(\tilde{w}_s^{\dagger}(\mathbf{s}')|\tilde{w}_s(S))),$$

the covariance function of $\tilde{w}_s^{\dagger}(\mathbf{s})$ is

$$\tilde{\mathcal{C}}_s^{\dagger}(\mathbf{s},\mathbf{s}') = \begin{cases} \Sigma_{\mathbb{Y}}^{\dagger}(\mathbf{s},\mathbf{s}'), & \text{if } \mathbf{s},\mathbf{s}'\in S;\\ -B_{\mathbf{s}}\Sigma_{\mathbb{Y}}^{\dagger}(S,\mathbf{s}'), & \text{if } \mathbf{s}\notin S,\ \mathbf{s}'\in S;\\ B_{\mathbf{s}}\Sigma_{\mathbb{Y}}^{\dagger}B_{\mathbf{s}'}^{T}, & \text{if } \mathbf{s},\mathbf{s}'\notin S,\ \mathbf{s},\mathbf{s}' \text{ belong to different blocks;}\\ B_{\mathbf{s}}\Sigma_{\mathbb{Y}}^{\dagger}B_{\mathbf{s}'}^{T}+\Sigma_{con}^{(p_k)}(\mathbf{s},\mathbf{s}'), & \text{if } \mathbf{s},\mathbf{s}'\notin S,\ \mathbf{s},\mathbf{s}' \text{ belong to the same block } p_k, \end{cases}$$

where $\Sigma_{\mathbb{Y}}^{\dagger}(S_1,S_2)$ and $\Sigma_{con}^{(p_k)}(S_1,S_2)$ denote the sub-matrices of $\Sigma_{\mathbb{Y}}^{\dagger}$ and $\Sigma_{con}^{(p_k)}$ corresponding to the covariance of location sets S_1 and S_2, respectively. The covariance function of the data process by the SFSA approach is

$$\mathcal{C}^{\dagger}(\mathbf{s},\mathbf{s}') = \mathcal{C}_l(\mathbf{s},\mathbf{s}') + \tilde{\mathcal{C}}_s^{\dagger}(\mathbf{s},\mathbf{s}').$$

So the approximated cross covariance between the predictive set S_p and the training set S is $U_p\mathcal{C}_*U^T - B_p\Sigma_{\mathbb{Y}}^{\dagger}$ and the kriging formula yields the conditional mean of $\mathbf{Y}_p$ given $\mathbb{Y}$ in Theorem 2.3.1. The conditional variance in Theorem 2.3.1 can be obtained similarly.

2.4 Simulation Study

In this section, we illustrate the effectiveness of our method through simulation studies and a precipitation dataset. The implementations of all methods are written in Matlab and run on a processor with 2.9 GHz Xeon CPUs and 16GB memory. For likelihood function optimization, we use the matlab function fminunc which implements a Broyden-Fletcher-Goldfarb-Shanno (BFGS) based Quasi-Newton method.

2.4.1 Prediction in spatial holes

Consider 4000 locations randomly selected in the spatial domain $\mathcal{S} = [0, 10] \times [0, 10]$. We held out 208 locations in the rectangular regions $[1.5, 3.5] \times [1.5, 2.5]$ and $[6.5, 8.5] \times [6.5, 7.5]$ as the predictive locations, and the rest of locations were used as training locations. This prediction scenario mimics the missing pattern of the satellite datasets where missing data are usually due to sizable holes. We generate realizations of $Y(\mathbf{s})$ from the model (3.1) with $\boldsymbol{\beta} = 0$ and the Matérn covariance model (2.2). A nugget effect τ^2 is added to covariance model (2.2) accounting for the measurement errors of responses. We compare the proposed method (denoted by "SFSA") with the FSA-Block approach (denoted by "FSAB"), and the block version of the conditional likelihood approximation approach in [18] (denoted by "CCL") in terms of parameter estimation and prediction results. The full covariance model results (denoted by "FullModel") serve as the baseline. All three methods require a partition of the observed dataset, and the SFSA and CCL approaches also need an ordering of data blocks. In this simulation study, the regular grid block centers are created for the block partition and the block numbers are sorted in an increasing order from northwest corner to southeast corner. For the SFSA and the FSA-Block approaches, we randomly select $m = 50, 100$ knot locations in $\mathcal{S}$. For the SFSA and the CCL approaches, we consider using the nearest block ($q = 1$) in the past blocks as the neighbor of each block. The Maximum Likelihood Estimates (MLEs) of model parameters are calculated based on the training set and the Mean Squared Prediction Errors (MSPEs) are computed for the prediction set. The Negative Log-Likelihood values (NLL) up to a constant $\frac{n}{2}\log(2\pi)$ are also obtained as a measure of model fitting. We experiment different parameter values of the covariance model (2.2) and the results are shown in Table 2.1.

In terms of parameter estimation, the SFSA always produces close means and Mean Squared Errors (MSE) to the full model results for covariance parameters under different dependence range scenarios. For the exponential model $\nu = 0.5$, when the dependence range is relatively small, both the SFSA and the CCL methods have smaller MSEs than that by the FSA-Block approach, this is because the predictive process component of the FSA-Block approach doesn't give an adequate approximation to the dependence across blocks, and the loss of information is relatively severe under the small-scale dependence scenario; when the dependence range is relatively large such as $\phi = 4$, both the SFSA and the FSA-Block outperform the CCL method, since the correlations across blocks are not negligible. In this case, the predictive process component does a good job in approximating the cross-block dependence, so the SFSA and the FSA-Block can produce smaller MSEs of model parameters. We remark that the SFSA approach produces smaller MSEs and NLLs than that by the FSA-Block approach in all 3 dependence range cases for the exponential model, due to the additional corrections of correlations among neighboring blocks. In addition, the parameter estimation results of the SFSA approach are less sensitive to the knot numbers compared to the FSA-Block approach. For the Matérn covariance model with $\nu = 1$, we have similar observations to the exponential model case. When the knot number is small ($m = 50$), the FSA-Block approach yields significantly larger MSEs than that of the SFSA method for parameters σ^2 and ϕ; the proposed

Table 2.1: Means and Mean Squared Errors (in parenthesis) of parameters in Matérn covariance model. The number of blocks $K = 100$ for all three methods and the results are based on 200 simulated datasets.

	$\sigma^2(1)$	$\phi(1)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.01(0.037)	1.04(0.045)	–	$0.011(6.12 \cdot 10^{-6})$	-1471.32	0.336
$FSAB, m = 100$	1.01(0.031)	1.05(0.040)	–	$0.011(6.86 \cdot 10^{-6})$	-1506.92	0.314
CCL	0.99(0.025)	1.00(0.028)	–	$0.010(4.33 \cdot 10^{-6})$	-1580.48	0.336
$SFSA, m = 50$	1.00(0.027)	1.01(0.030)	–	$0.010(4.47 \cdot 10^{-6})$	-1599.30	0.319
$SFSA, m = 100$	1.00(**0.025**)	1.02(**0.029**)	–	$0.011(4.67 \cdot 10^{-6})$	$\mathbf{-1613.68}$	**0.304**
$FullModel$	0.99(0.021)	1.00(0.024)	–	$0.010(4.20 \cdot 10^{-6})$	-1690.62	0.294
	$\sigma^2(1)$	$\phi(2)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.04(0.155)	2.14(0.697)	–	$0.011(2.67 \cdot 10^{-6})$	-2565.50	0.182
$FSAB, m = 100$	1.04(0.091)	2.15(0.422)	–	$0.011(2.92 \cdot 10^{-6})$	-2609.06	0.172
CCL	1.00(0.091)	2.00(0.375)	–	$0.010(2.16 \cdot 10^{-6})$	-2664.69	0.186
$SFSA, m = 50$	1.01(0.086)	2.04(0.363)	–	$0.010(2.18 \cdot 10^{-6})$	-2695.87	0.174
$SFSA, m = 100$	1.01(**0.081**)	2.05(**0.351**)	–	$0.010(2.19 \cdot 10^{-6})$	$\mathbf{-2713.50}$	**0.168**
$FullModel$	1.00(0.069)	2.00(0.294)	–	$0.010(2.13 \cdot 10^{-6})$	-2785.12	0.156
	$\sigma^2(1)$	$\phi(4)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.05(0.242)	4.36(4.21)	–	$0.010(1.21 \cdot 10^{-6})$	-3568.62	0.099
$FSAB, m = 100$	1.03(0.185)	4.28(3.32)	–	$0.010(1.16 \cdot 10^{-6})$	-3609.01	0.094
CCL	1.02(0.462)	4.09(7.00)	–	$0.010(9.27 \cdot 10^{-7})$	-3654.54	0.106
$SFSA, m = 50$	1.00(0.166)	4.04(2.71)	–	$0.010(9.09 \cdot 10^{-7})$	-3692.02	0.095
$SFSA, m = 100$	0.99(**0.138**)	4.00(**2.29**)	–	$0.010(8.76 \cdot 10^{-7})$	$\mathbf{-3704.48}$	**0.092**
$FullModel$	0.98(0.134)	3.91(2.14)	–	$0.010(8.97 \cdot 10^{-7})$	-3776.61	0.088
	$\sigma^2(1)$	$\phi(1)$	$\nu(1)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.04(0.117)	1.05(0.074)	1.00(0.0055)	$0.010(1.09 \cdot 10^{-9})$	-4228.07	0.099
$FSAB, m = 100$	1.01(0.096)	1.02(0.064)	1.01(0.0050)	$0.010(5.28 \cdot 10^{-8})$	-4304.52	0.086
CCL	0.98(0.068)	0.98(0.037)	1.01(0.0046)	$0.010(4.82 \cdot 10^{-10})$	-4403.75	0.106
$SFSA, m = 50$	1.00(0.069)	1.00(0.036)	1.01(0.0043)	$0.010(3.81 \cdot 10^{-10})$	-4443.47	0.093
$SFSA, m = 100$	0.98(**0.063**)	0.98(**0.034**)	1.01(**0.0042**)	$0.010(7.80 \cdot 10^{-9})$	$\mathbf{-4473.19}$	**0.082**
$FullModel$	0.98(0.055)	0.98(0.030)	1.01(0.0035)	$0.010(2.90 \cdot 10^{-10})$	-4588.23	0.075
	$\sigma^2(1)$	$\phi(2)$	$\nu(1)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.33(0.951)	2.40(1.240)	1.00(0.0062)	$0.010(3.17 \cdot 10^{-12})$	-5316.19	0.040
$FSAB, m = 100$	1.07(0.183)	2.07(0.358)	1.03(0.0065)	$0.010(1.81 \cdot 10^{-8})$	-5396.27	0.035
CCL	1.08(0.238)	2.10(0.448)	1.00(0.0047)	$0.010(1.41 \cdot 10^{-8})$	-5419.63	0.042
$SFSA, m = 50$	1.12(0.261)	2.14(0.438)	1.00(**0.0040**)	$0.010(4.92 \cdot 10^{-9})$	-5480.01	0.037
$SFSA, m = 100$	1.07(**0.190**)	2.07(**0.340**)	1.01(0.0042)	$0.010(5.53 \cdot 10^{-10})$	$\mathbf{-5509.77}$	**0.034**
$FullModel$	1.07(0.166)	2.08(0.294)	1.00(0.0034)	$0.010(1.24 \cdot 10^{-8})$	-5584.77	0.030

method produces similar MSEs of parameters for different knot numbers, indicating its insensitiveness to the knot number.

In terms of prediction, the SFSA and FSA-Block approach outperform the CCL approach in all scenarios, this may be because the knot-based approaches can gain

Table 2.2: MSPEs and MSEs of the exponential model with sample size 4000. The number of blocks $K = 100$.

	$\sigma^2(1)$	$\phi(1)$	$\tau^2(0.2)$	NLL	MSPE
$FSAB, m = 50$	1.00(0.035)	1.07(0.066)	$0.204(1.45 \cdot 10^{-4})$	292.93	0.575
$FSAB, m = 100$	1.00(0.032)	1.09(0.062)	$0.205(1.39 \cdot 10^{-4})$	271.26	0.552
CCL	0.99(0.029)	0.99(0.043)	$0.198(1.16 \cdot 10^{-4})$	243.44	0.573
$SFSA, m = 50$	0.98(0.026)	1.00(0.041)	$0.200(1.13 \cdot 10^{-4})$	224.60	0.563
$SFSA, m = 100$	0.99(0.026)	1.02(0.041)	$0.201(1.11 \cdot 10^{-4})$	216.81	0.544
$FullModel$	0.98(0.025)	0.98(0.035)	$0.199(1.0 \cdot 10^{-4})$	174.39	0.525
	$\sigma^2(1)$	$\phi(2)$	$\tau^2(0.2)$	NLL	MSPE
$FSAB, m = 50$	1.09(0.219)	2.43(1.460)	$0.204(7.67 \cdot 10^{-5})$	-150.74	0.410
$FSAB, m = 100$	1.07(0.141)	2.40(0.993)	$0.204(7.20 \cdot 10^{-5})$	-181.75	0.398
CCL	0.98(0.088)	1.99(0.423)	$0.199(5.78 \cdot 10^{-5})$	-184.00	0.419
$SFSA, m = 50$	1.02(0.102)	2.13(0.552)	$0.201(5.61 \cdot 10^{-5})$	-210.51	0.404
$SFSA, m = 100$	1.01(0.080)	2.12(0.446)	$0.201(5.50 \cdot 10^{-5})$	-223.05	0.393
$FullModel$	0.98(0.072)	1.98(0.354)	$0.199(5.38 \cdot 10^{-5})$	-252.81	0.377
	$\sigma^2(1)$	$\phi(4)$	$\tau^2(0.2)$	NLL	MSPE
$FSAB, m = 50$	1.15(0.518)	5.25(12.30)	$0.202(5.26 \cdot 10^{-5})$	-481.89	0.316
$FSAB, m = 100$	1.02(0.179)	4.60(4.03)	$0.202(4.79 \cdot 10^{-5})$	-512.48	0.309
CCL	0.94(0.275)	3.8(4.82)	$0.198(5.04 \cdot 10^{-5})$	-499.95	0.324
$SFSA, m = 50$	0.99(0.214)	4.18(3.80)	$0.200(4.39 \cdot 10^{-5})$	-528.58	0.311
$SFSA, m = 100$	0.96(0.150)	4.07(2.72)	$0.200(4.37 \cdot 10^{-5})$	-538.93	0.305
$FullModel$	0.93(0.131)	3.74(2.24)	$0.198(4.45 \cdot 10^{-5})$	-561.92	0.300

additional information for the predictive locations in the spatial hole when we place knots around or in the spatial hole. The prediction errors of all 3 methods decrease with the increase of dependence range parameter ϕ, since more information can be borrowed from the training set under the large scale dependence structure. Compared with the FSA-Block approach with equal number of knots, the SFSA has smaller MSPEs due to additional corrections of correlations across blocks. The prediction difference between SFSA and FSA-Block is relatively larger when the dependence scale is small and the process is less smooth, since in this case, the predictive process component of the FSA-Block doesn't perform well in approximating the correlations across blocks. Table 2.2 shows the parameter estimation and prediction results when the data is generated with a high noise level 0.2. The observations are similar to the simulation set-up with a low noise level 0.01.

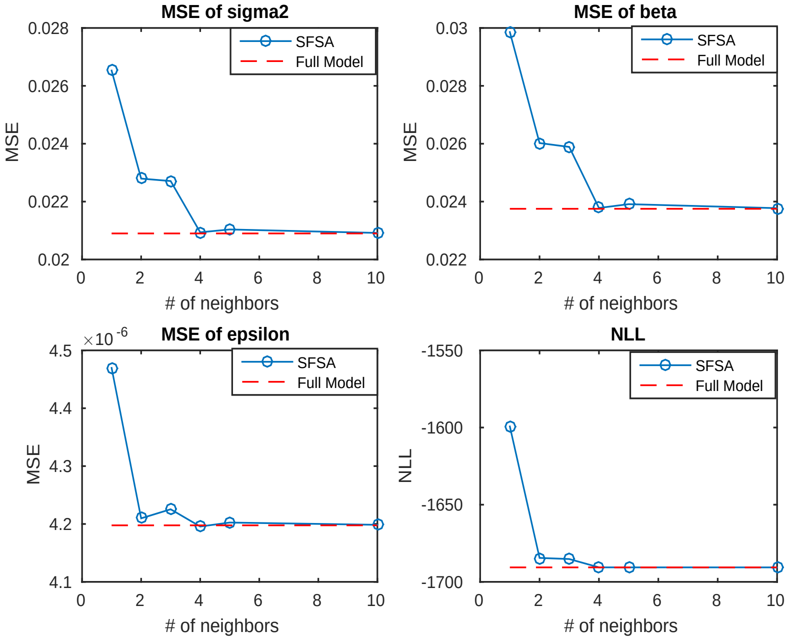


Figure 2.2: MSE of each covariance parameter against the number of neighboring blocks for the SFSA approach. The simulation setting is the same as that described in Section 2.4.1; the covariance model is exponential model with $\phi = 1$, $\sigma^2 = 1$, and $\tau^2 = 0.01$.

Figure 2.2 shows how the MSEs and NLL change with the number of neighboring blocks for the SFSA approach. When $q \geq 4$, the SFSA approach can produce results almost at par with the full covariance model. Notice that the neighboring locations of a block for $q = 4$ is only about 200, so the proposed method can achieve good parameter estimation results at very low costs of computations. For predictions, figure 2.3 shows how the MSPEs change with number of neighboring blocks. We find that the prediction performance is more sensitive to the block size than to the number of neighbors for the prediction in spatial hole scenario. This may be because for the grid partition, each spatial hole covers more than one block when we choose K to be large. In this case the predictive locations belonging to the same block can not have enough nearby locations due to the block ordering. Thus when we partition the data into small number of blocks, the prediction error drops more significantly due to the increased number of very close locations in the neighbor set.

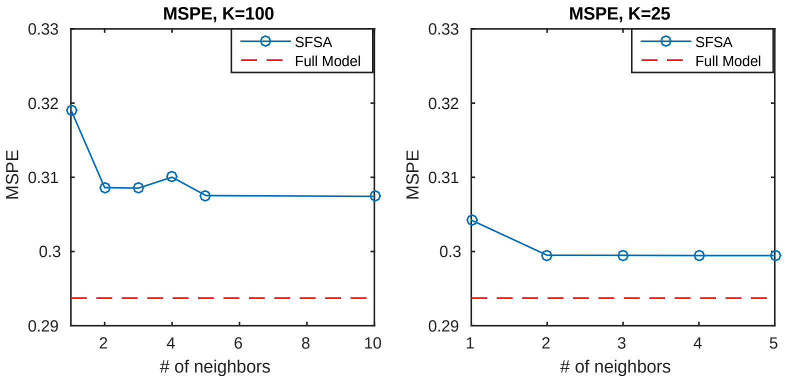


Figure 2.3: MSPEs against the number of neighboring blocks for the SFSA approach with $K = 25, 100$. The simulation setting is the same as that described in Section 2.4.1; the covariance model is exponential model with $\phi = 1$, $\sigma^2 = 1$, and $\tau^2 = 0.01$.

2.4.2 Prediction on block boundaries

For the FSA-Block approach, when predicting responses on shared boundaries of two blocks, the prediction of a boundary location depends on which block it belongs to. Although the FSA-Block approach has the predictive process part as the "global" support, the discontinuity caused by the independent blocks approximation of the residual covariance can be severe when the predictive process component does not perform well due to limited knot number. Since the proposed approach takes the neighboring blocks information into account for modeling the residual process, we expect that it can produce a smoother prediction surface than that by the FSA-Block approach. Compared with the conditional likelihood approximation method [18], it may also relieve the discontinuity problem for predicting on boundary locations due to the additional global information provided by the low-rank process part. In the following simulation example, we compare the prediction performance of three methods on locations of block boundaries. We use the regular grid to create the same 100 data blocks as in the previous example, so the lines $s_1 = 1, 2, \cdots, 9$ and $s_2 = 1, 2, \cdots, 9$ constitute the boundaries of the block partition. We randomly generated 4000 training locations in the spatial domain $\mathcal{S}$. Then 20 predictive locations were randomly selected on each of the 18 boundary lines to form a prediction set of 360 locations. The ordering of blocks and knot selection method are the same as in the previous example. We experimented different parameter values of covariance model (2.2) with $\nu = 0.5$ and the results are shown in Table 2.3.

We focus on the prediction performances of three competing methods. In all 3 scenarios of dependence ranges for the exponential model, the SFSA approach produces the smallest MSPEs on block boundary locations. The prediction performance of CCL method is close to that of the SFSA method, indicating that borrowing information from neighboring points for prediction can create a relatively smoother prediction surface. The FSA-Block method does not perform as well as the SFSA method on predicting boundary locations; especially in the case of relatively small dependence range, the prediction errors by the FSA-Block method is significantly larger than that by the SFSA approach. For the Matérn covariance with $\nu = 1$, the SFSA approach still outperforms the FSA-Block in terms of prediction error of boundary locations. But the differences of prediction errors by the two approaches are smaller, since the predictive process component serving as a global predictor performs better when the underlying process is smoother. We also experimented the scenario of the responses with a relatively large nugget and results are shown in Table 2.4. In this scenario, the SFSA and CCL methods outperform the FSA-Block method for different dependence ranges; the differences of MSPEs between the SF-

Table 2.3: MSPEs and MSEs (in parenthesis) of the exponential model with sample size 4000. 360 boundary locations were held out for prediction. The number of blocks $K = 100$.

	$\sigma^2(1)$	$\phi(1)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.02(0.039)	1.06(0.050)	–	$0.011(7.20 \cdot 10^{-6})$	-1580.77	0.168
$FSAB, m = 100$	1.02(0.028)	1.06(0.035)	–	$0.011(7.77 \cdot 10^{-6})$	-1613.39	0.160
CCL	1.01(0.027)	1.02(0.030)	–	$0.010(5.35 \cdot 10^{-6})$	-1691.35	0.139
$SFSA, m = 50$	1.01(0.028)	1.03(0.031)	–	$0.010(5.46 \cdot 10^{-6})$	-1711.95	0.137
$SFSA, m = 100$	1.01(0.024)	1.03(0.027)	–	$0.010(5.64 \cdot 10^{-6})$	-1725.45	**0.133**
$FullModel$	1.01(0.023)	1.02(0.025)	–	$0.010(5.04 \cdot 10^{-6})$	-1812.47	0.104
	$\sigma^2(1)$	$\phi(2)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.06(0.160)	2.19(0.734)	–	$0.011(2.90 \cdot 10^{-6})$	-2724.91	0.091
$FSAB, m = 100$	1.08(0.106)	2.25(0.508)	–	$0.011(2.90 \cdot 10^{-6})$	-2774.28	0.087
CCL	1.03(0.103)	2.07(0.437)	–	$0.010(2.07 \cdot 10^{-6})$	-2829.41	0.078
$SFSA, m = 50$	1.03(0.085)	2.09(0.367)	–	$0.010(2.11 \cdot 10^{-6})$	-2861.10	0.075
$SFSA, m = 100$	1.04(0.078)	2.13(0.339)	–	$0.010(2.10 \cdot 10^{-6})$	-2881.34	**0.074**
$FullModel$	1.02(0.062)	2.05(0.259)	–	$0.010(1.85 \cdot 10^{-6})$	-2962.68	0.059
	$\sigma^2(1)$	$\phi(4)$	$\nu(0.5)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.04(0.245)	4.32(4.29)	–	$0.010(9.97 \cdot 10^{-7})$	-3782.32	0.053
$FSAB, m = 100$	1.03(0.209)	4.27(3.70)	–	$0.010(1.04 \cdot 10^{-6})$	-3825.53	0.051
CCL	1.01(0.273)	4.03(4.36)	–	$0.010(8.08 \cdot 10^{-7})$	-3872.83	0.046
$SFSA, m = 50$	0.97(0.131)	3.93(2.12)	–	$0.010(7.89 \cdot 10^{-7})$	-3910.54	0.044
$SFSA, m = 100$	0.98(0.126)	3.99(2.06)	–	$0.010(7.99 \cdot 10^{-7})$	-3925.72	**0.044**
$FullModel$	0.97(0.107)	3.87(1.67)	–	$0.010(7.38 \cdot 10^{-7})$	-4004.71	0.036
	$\sigma^2(1)$	$\phi(1)$	$\nu(1)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.03(0.139)	1.06(0.081)	0.99(0.0052)	$0.010(6.79 \cdot 10^{-11})$	-4491.28	0.037
$FSAB, m = 100$	1.04(0.108)	1.06(0.054)	1.00(0.0044)	$0.010(1.21 \cdot 10^{-8})$	-4560.49	0.034
CCL	1.01(0.083)	1.01(0.040)	1.00(0.0033)	$0.010(1.47 \cdot 10^{-9})$	-4667.55	0.029
$SFSA, m = 50$	1.00(0.078)	1.02(0.041)	1.00(0.0035)	$0.010(1.48 \cdot 10^{-11})$	-4711.32	0.028
$SFSA, m = 100$	1.01(0.079)	1.02(0.035)	1.00(0.0033)	$0.010(1.57 \cdot 10^{-9})$	-4738.50	**0.027**
$FullModel$	1.00(0.065)	1.01(0.030)	1.00(0.0029)	$0.010(9.97 \cdot 10^{-10})$	-4867.43	0.020
	$\sigma^2(1)$	$\phi(2)$	$\nu(1)$	$\tau^2(0.01)$	NLL	MSPE
$FSAB, m = 50$	1.16(0.504)	2.17(0.709)	1.02(0.0069)	$0.010(1.68 \cdot 10^{-8})$	-5616.22	0.0199
$FSAB, m = 100$	1.04(0.168)	1.99(0.303)	1.04(0.0074)	$0.010(4.84 \cdot 10^{-8})$	-5701.62	0.0186
CCL	1.02(0.223)	1.98(0.371)	1.01(0.0050)	$0.010(1.60 \cdot 10^{-10})$	-5728.09	0.0178
$SFSA, m = 50$	1.05(0.217)	2.01(0.327)	1.02(0.0044)	$0.010(1.47 \cdot 10^{-9})$	-5789.06	0.0169
$SFSA, m = 100$	1.05(0.221)	2.01(0.343)	1.02(0.0048)	$0.010(3.51 \cdot 10^{-9})$	-5820.22	**0.0165**
$FullModel$	1.03(0.195)	1.99(0.304)	1.01(0.0040)	$0.010(3.49 \cdot 10^{-10})$	-5903.00	0.0143

SA approach and the FSA-Block approach are larger than the small-nugget scenario when both the knot number and the dependence range are small.

Table 2.4: MSPEs of the exponential model with $\sigma^2 = 1$ and a larger nugget $\tau^2 = 0.2$. 360 boundary locations were held out for prediction. The number of blocks $K = 100$.

Exponential	$FSAB, m = 50$	$m = 100$	CCL	$SFSA, m = 50$	$m = 100$	$FullModel$
$\phi = 1$	0.404	0.396	0.377	0.370	**0.367**	0.333
$\phi = 2$	0.320	0.311	0.306	0.299	**0.296**	0.279
$\phi = 4$	0.270	0.265	0.263	0.259	**0.257**	0.246

2.5 Real Data Analysis

We analyze the precipitation dataset in United States for the years from 1895 to 1997. It was collected by National Climate Data Center and we consider the yearly total precipitation anomalies in this analysis, which are yearly totals standardized by the long-run mean and standard deviation of each weather station. We select the precipitation anomalies in 1962 [43] to illustrate the proposed SFSA approach, since this year contains the largest number of observations. According to the analysis in [38], this dataset appears no significant nonstationarity and anisotropy, therefore the isotropic covariance models can be applied in the spatial regression model (3.1) for fitting this dataset. Since observations are on the sphere, the chordal distance with unit in kilometers is used to calculate the distances among weather stations to ensure positive-definiteness of the covariance function.

We first partitioned the total 7352 observations into a training set of 7000 observations and a prediction set of 352 observations. The prediction set contains 143 locations in the space hole $(-87, -82) \times (35, 38)$ and 209 locations randomly selected in the rest of the area. The covariance model (2.2) was used to model the spatial dependence among observations of weather stations. We fixed the smoothness parameter $\nu = 0.5$, so the covariance model is the exponential covariance function. The K-means clustering algorithm was applied to the training set for creating data blocks; then we ordered the created data blocks. We also applied K-means clustering algorithm on the training dataset to obtain 300 cluster centers for the knot set. For both the SFSA and the CCL methods, the neighboring set of a block is its nearest neighboring data block. The following Table 2.5 shows the parameter estimation and prediction results of three approaches by both the Maximum likelihood and the Bayesian methods; the results of the full covariance model serve as the baseline. For the Bayesian inference, we obtain the Maximum A Posteriori (MAP) estimates of parameters and the corresponding MSPEs by using the MAP estimates.

Table 2.5: Maximum Likelihood and Bayesian inference results using the exponential model.

MLEs	σ^2	ϕ	τ^2	Log lik	MSPE
$FSAB, K = 70$	0.6792	180.22	0.1078	-5218.69	0.311
$CCL, K = 70$	0.6866	174.37	0.1043	-5206.07	0.329
$SFSA, K = 70$	0.6734	170.96	0.1045	$\mathbf{-5179.15}$	**0.301**
$FSAB, K = 25$	0.6780	172.29	0.1045	-5190.36	0.296
$CCL, K = 25$	0.6874	169.64	0.1020	-5177.63	0.282
$SFSA, K = 25$	0.6863	170.86	0.1026	$\mathbf{-5160.71}$	**0.274**
$Full\ Model$	0.6757	166.84	0.1023	-5150.60	0.272
Bayesian	σ^2	ϕ	τ^2	DIC	MSPE
$FSAB, K = 70$	0.6718	177.85	0.1074	10439.50	0.309
$CLL, K = 70$	0.6903	174.68	0.1039	10418.12	0.329
$SFSA, K = 70$	0.6667	168.22	0.1040	**10357.64**	**0.308**
$FSAB, K = 25$	0.6707	171.07	0.1050	10392.52	0.300
$CLL, K = 25$	0.6878	170.40	0.1024	10361.55	0.281
$SFSA, K = 25$	0.6834	171.82	0.1036	**10329.95**	**0.276**
$Full\ Model$	0.6706	165.65	0.1024	10307.03	0.272

The MLEs of model parameters by all three methods are close to the full model results. The SFSA approach produces the largest log-likelihood among 3 competing methods for the same block partition, since it includes other two methods as special cases. In terms of prediction, when the block number K is relatively large, the prediction errors of the SFSA and the FSA-Block methods are significantly smaller than that of the CCL approach, since the additional correction of residual covariance between one block and its nearest neighbor may not be as crucial as the inclusion of the predictive process component. When the block number K is relatively small, the prediction errors of the SFSA and the CCL methods are significantly smaller than that of the FSA-Block approach, since the additional correction of residual covariance may be more crucial for the case of larger block size. We remark that the SFSA approach has the smallest MSPE for a given K and it has very close prediction performance to the full model when $K = 25$. The Bayesian inference results are very similar to that of MLEs. We can observe that the MAP estimate of each model parameter by the SFSA approach is close to estimate by the full covariance model. Besides, the Deviance Information Criteria (DIC) score by the SFSA approach is smallest among three methods, indicating that it fits the data best.

2.6 Discussion

We propose a Smooth Full-Scale Approximation approach (SFSA) which extends the FSA-Block approach by correcting the approximation errors of correlations among neighboring blocks. The proposed method incorporates the FSA-Block approach and the block conditional likelihood approach as special cases and can achieve better statistical efficiency. Compared with the FSA-Block approach, the SFSA approach can significantly alleviate the discontinuity problem of predictions on boundary locations and produce a smoother prediction surface. In addition, due to the additional corrections of correlations across data blocks, the SFSA approach is less sensitive to the knot set than the FSA-Block approach. So it can produce more robust results when the number of knots is not sufficient or the knot locations are not properly placed. The computational complexity of the proposed method is linear with sample size n when the knot number m, block size n_b, and number of neighboring blocks q are small. Numerical results show that for a few hundreds of knots, block size, and neighboring locations, the SFSA approach can achieve very close parameter estimation and prediction results to the full covariance model. Therefore, it can serve as a computationally efficient tool for modeling large spatial datasets.

3. FULL-SCALE APPROXIMATIONS OF SPATIO-TEMPORAL COVARIANCE MODELS*

3.1 Introduction

Spatio-temporal datasets are widely collected in many scientific disciplines, where the data are observed in both space and time. The primary interests in analyzing spatio-temporal datasets are to detect meaningful spatio-temporal dependence patterns, and to subsequently smooth and predict in space-time domain. Recent developments are mainly in spatio-temporal process models. There are two major paradigms for modeling dependence structures of spatio-temporal datasets. The first treats time as discrete and views data as time series of spatial process realizations. Many works [24, 14, 42, 20] following this direction adopt a state-space framework, where the dynamics in a time series of spatial fields are explained by a sequence of state variables driven by a stochastic process.

We focus on a paradigm in which both space and time are continuously indexed. A key ingredient is a valid spatio-temporal covariance model that characterizes spatio-temporal dependence structure. A simple but commonly used class of spatio-temporal covariance model assumes a separable form that factors into a purely spatial and a purely temporal component. However, separable models do not allow for space-time interaction and often fail to model a physical process adequately. There is a growing literature on methods for constructing more flexible spatio-temporal covariance functions. [12] introduced several classes of nonseparable spatio-temporal covariance functions based on the spectral density of nonnegative

*Reprinted with permission from “Full-scale approximations of spatio-temporal covariance models for large datasets” by Bohai Zhang, Huiyan Sang, and Jianhua Z Huang, 2015. *Statistica Sinica*, 25, 99-114, Copyright [2015] by Academia Sinica.

finite measures. [30] extended their work and introduced a broader class of nonseparable spatio-temporal covariance functions, which does not depend on closed forms of inverse Fourier transforms. [66] developed a class of asymmetric and nonstationary space-time covariance functions on the sphere.

Parameter estimation and spatio-temporal prediction with these models typically require $\mathcal{O}(n^3)$ operations for a spatio-temporal dataset of size n, imposing computational challenges. One approach to the computation seeks to simplify the model fitting method mainly through likelihood approximations. Composite likelihood (CL) methods [71] have been applied to deal with spatial and spatio-temporal datasets due to their simplicity and sound asymptotic properties. The idea is to use a pseudo-likelihood by combining likelihood objects constructed from conditional or marginal models as a surrogate for the ordinary likelihood. Recently, [6] introduced a weighted composite likelihood (WCL) method based on pairwise differences of spatio-temporal observations. They showed that the estimators of their methods are consistent and asymptotic normal under increasing domain asymptotics [11]. [3] also developed a CL method based on pairwise differences, forming a joint estimation function based on spatial, temporal and spatio-temporal group-based estimation functions. A second approach seeks to simplify model specifications of covariance structures to achieve computational efficiency. Many literature following this path have emerged but primarily focusing on spatial or spatial discrete-time contexts. Existing works here include covariance tapering [23, 43], Gaussian Markov random-field approximation [56, 57] and reduced rank approximation [37, 77, 73, 39, 13, 42]. [4] proposed a class of spatial predictive processes models that is applicable to fitting spatio-temporal process models with large data sets. The idea of this reduced rank type of approach is to approximate a spatio-temporal process with a predictive process, the prediction of a given spatio-temporal process conditional on the random spatio-temporal vector at a selected location set of reduced size.

Recently, [59] developed a covariance approximation method, referred to as *full-scale approximation* (FSA), for the implementation of univariate spatial process models with large datasets. Combining merits of reduced rank techniques [13, 4] and sparse matrix algorithms [23], the FSA approach provides a high quality approximation to the covariance function at both large and small spatial scales and achieves computational efficiency.

In this Section, we extend the FSA approach to the spatio-temporal process context. We propose a general-purpose full-scale approximation that can approximate well the original covariance function at both large and small spatio-temporal scales. Here the first step produces a reduced rank spatio-temporal covariance that is effective in capturing large-scale spatio-temporal dependence and the second step a sparse covariance matrix that can approximate well the small-scale spatio-temporal dependence unexplained by the first part. Our method yields a new full-scale spatio-temporal covariance function for any given covariance function that maintains the flexibility and the richness of spatio-temporal structure while substantially reducing computational cost. Spatio-temporal predictions of the full-scale covariance approximation models can be carried out efficiently following the conventional prediction procedure in Gaussian processes.

The method requires careful selection of knots in the reduced rank step, an issue not addressed by [59], to achieve a good balance between model fitting and computational time. We propose an adaptive and automatic way to select both the knot number and knot locations by treating them as random variables. We consider selecting knots either from a set containing all spatio-temporally observed locations or a fine grid covering entire space-time domain following a Reversible Jump Markov chain Monte Carlo (RJMCMC) algorithm [33].

3.2 The FSA Approach

3.2.1 Model

Let $Y(\mathbf{s}, t)$ be a response variable observed at location $\mathbf{s}$ and time t, where $\mathbf{s} \in \mathcal{S} \subseteq \mathbb{R}^2,\ t \in [0,\ T] \subseteq \mathbb{R}^+$. A general formulation of spatio-temporal process model is

$$Y(\mathbf{s}, t) = \mu(\mathbf{s}, t) + w(\mathbf{s}, t) + \epsilon(\mathbf{s}, t), \tag{3.1}$$

where $\mu(\mathbf{s}, t)$ is a deterministic mean function, $w(\mathbf{s}, t)$ is a zero-mean Gaussian process characterizing spatio-temporal variations, and $\epsilon(\mathbf{s}, t)$ is a Gaussian white noise process independent of $w(\mathbf{s}, t)$. The variance of $\epsilon(\mathbf{s}, t)$ is usually assumed to be a constant τ^2, called the "nugget effect", to account for measurement errors. The spatio-temporal dependence structure of $w(\mathbf{s}, t)$ is specified by a spatio-temporal covariance function $\Gamma_w(\mathbf{s}, t; \mathbf{s}', t') \equiv \mathrm{Cov}(w(\mathbf{s}, t), w(\mathbf{s}', t'))$. In the spatio-temporal regression framework, we assume $\mu(\mathbf{s}, t) = Z^T(\mathbf{s}, t)\boldsymbol{\beta}$, where $Z(\mathbf{s}, t)$ is a $p \times 1$ column vector of space-time covariates and $\boldsymbol{\beta}$ is the associated coefficient vector.

To simplify notation, denote a spatio-temporal point $\mathbf{x}$ by $(\mathbf{s}, t)$. Let $\mathscr{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$, $\mathbf{w} = (w(\mathbf{x}_1), w(\mathbf{x}_2), \cdots, w(\mathbf{x}_n))^T$ and $\boldsymbol{\epsilon} = (\epsilon(\mathbf{x}_1), \epsilon(\mathbf{x}_2), \cdots, \epsilon(\mathbf{x}_n))^T$. It follows that $\mathbf{w} \sim \mathrm{MVN}(\mathbf{0}, \Sigma_{\mathbf{w}})$ and $\boldsymbol{\epsilon} \sim \mathrm{MVN}(\mathbf{0}, \Sigma_{\boldsymbol{\epsilon}})$, where MVN stands for the multivariate normal distribution, $\Sigma_{\mathbf{w}} = [\Gamma_{\mathbf{w}}(\mathbf{x}_i, \mathbf{x}_j)]_{i=1:n, j=1:n}$, and $\Sigma_{\boldsymbol{\epsilon}} = \tau^2\mathbf{I}_n$.

The marginal distribution of $\mathbb{Y} = (Y(\mathbf{x}_1), \cdots, Y(\mathbf{x}_n))^T \sim \mathrm{MVN}(\mathbb{Z}\boldsymbol{\beta}, \Sigma_{\mathbf{w}} + \tau^2\mathbf{I}_n)$, where $\mathbb{Z} = (Z(\mathbf{x}_1), Z(\mathbf{x}_2), \cdots, Z(\mathbf{x}_n))^T$. To make likelihood-based or Bayesian inferences, we need to evaluate the likelihood of $\mathbb{Y}$; this requires $\mathcal{O}(n^3)$ computational time due to the inversion of $\Sigma_{\mathbf{w}} + \tau^2\mathbf{I}_n$. A similar computational bottleneck is encountered when performing spatio-temporal prediction.

3.2.2 Covariance approximation of spatio-temporal process

We propose a full-scale covariance approximation for the latent spatio-temporal process w by a sum of two independent processes,

$$w^{\dagger}(\mathbf{x}) = w_l(\mathbf{x}) + w_s(\mathbf{x}), \tag{3.2}$$

where $w_l(\mathbf{x})$ is a low-rank process that captures the large-scale spatio-temporal dependence structure and $w_s(\mathbf{x})$ is a residual process that models the small-scale spatio-temporal dependence not captured by $w_l(\mathbf{x})$. We model the low rank process using a predictive process on spatio-temporal domain. The predictive process, proposed by [4], has been shown to be flexible enough to model the large-scale dependence structure of a spatial process. Given a set of points $\mathscr{X}^* = \{\mathbf{x}_1^*, \mathbf{x}_2^*, \cdots, \mathbf{x}_m^*\}$, called spatio-temporal knots, the spatio-temporal predictive process is

$$w_l(\mathbf{x}) = \mathcal{C}(\mathbf{x}, \mathscr{X}^*)\mathcal{C}^{*-1}\mathbf{w}^*,$$

where $\mathbf{w}^* = (w(\mathbf{x}_1^*), w(\mathbf{x}_2^*), \cdots, w(\mathbf{x}_m^*))^T$, $\mathcal{C}(\mathbf{x}, \mathscr{X}^*) = [\Gamma_w(\mathbf{x}, \mathbf{x}_j^*)]_{j=1:m}$, and $\mathcal{C}^* = \mathcal{C}(\mathscr{X}^*, \mathscr{X}^*)$ is the covariance matrix of $\mathbf{w}^*$. It follows that the covariance function of w_l is given by

$$\Gamma_{w_l}(\mathbf{x}, \mathbf{x}') = \mathcal{C}(\mathbf{x}, \mathscr{X}^*)\mathcal{C}^{*-1}\mathcal{C}^T(\mathbf{x}', \mathscr{X}^*). \tag{3.3}$$

The covariance matrix of w_l evaluated at $\mathscr{X}$ is $\Sigma_{w_l} = \mathcal{C}(\mathscr{X}, \mathscr{X}^*)\mathcal{C}^{*-1}\mathcal{C}^T(\mathscr{X}, \mathscr{X}^*)$ where $\mathcal{C}(\mathscr{X}, \mathscr{X}^*) = [\Gamma_w(\mathbf{x}_i, \mathbf{x}_j^*)]_{i=1:n,j=1:m}$. One often chooses $m \ll n$, which results in a low-rank matrix Σ_{w_l} and hence leads to efficient computations. If the knot set is chosen to be $\mathscr{X}$, the original spatio-temporal covariance is fully recovered.

The residual process $w_s(\mathbf{x})$ is an important supplement to $w_l(\mathbf{x})$ for better approximating the original process $w(\mathbf{x})$, while maintaining computational efficiency. The idea is to use a sparse covariance matrix to approximate the covariance of the exact residual process $w(\mathbf{x}) - w_l(\mathbf{x})$. The covariance function of $w(\mathbf{x}) - w_l(\mathbf{x})$ is $\Gamma_w(\mathbf{x}, \mathbf{x}') - \Gamma_{w_l}(\mathbf{x}, \mathbf{x}')$, and we take the covariance function of $w_s(\mathbf{x})$ to be

$$\Gamma_{w_s}(\mathbf{x}, \mathbf{x}') = \{\Gamma_w(\mathbf{x}, \mathbf{x}') - \Gamma_{w_l}(\mathbf{x}, \mathbf{x}')\}\mathcal{K}(\mathbf{x}, \mathbf{x}'), \tag{3.4}$$

where $\mathcal{K}(\mathbf{x}, \mathbf{x}')$, referred to as a modulating function, is chosen to ensure Γ_{w_s} is a valid positive semidefinite function and that is zero for a large proportion of possible spatio-temporal pairs $(\mathbf{x}, \mathbf{x}')$ evaluated at $\mathscr{X}$. The choice of $\mathcal{K}(\mathbf{x}, \mathbf{x}')$ ensures that the resulting residual covariance matrix Σ_{w_s} can be handled efficiently.

We describe several strategies for specifying the modulating function $\mathcal{K}$. The first is to use a tapering function, the result is referred to as *FSA-Taper*, which sets the correlation of distant spatio-temporal pairs to zero. In the univariate spatial case, a number of compactly supported covariance functions have been used for covariance tapering, for example the spherical covariance function, the family of Wendland covariance functions, and the bisquare function, to name a few [75, 76, 30, 13]. In the spatio-temporal context, we consider tapering functions as Schur products of a purely spatial and a purely temporal tapering function. Let $\mathcal{K}_u(\mathbf{s}, \mathbf{s}'; \gamma_u)$ be a tapering function on the spatial domain satisfying $\mathcal{K}_u = 0$ when $\|\mathbf{s} - \mathbf{s}'\| > \gamma_u$, and $\mathcal{K}_v(t, t'; \gamma_v)$ be a tapering function on the temporal domain such that $\mathcal{K}_v = 0$ when $\|t - t'\| > \gamma_v$. Here, γ_u and γ_v are referred to as the spatial taper range and the temporal taper range, respectively. Then $\mathcal{K}(\mathbf{s}, t; \mathbf{s}', t') = \mathcal{K}_u(\mathbf{s}, \mathbf{s}'; \gamma_u)\mathcal{K}_v(t, t'; \gamma_v)$.

A second specification of $\mathcal{K}$ uses local partitioning: residuals are assumed to be independent across partitioned space-time subregions, while the original residual covariance is preserved within each subregion. Let $B_1, B_2, \cdots, B_K$ be a partition of the space-time domain $\mathcal{S} \times [0,\ T]$, referred to as blocks. Then the modulating function is

$$\mathcal{K}_{block}(\mathbf{x}, \mathbf{x}') = \begin{cases} 1 & \text{if } \mathbf{x}, \mathbf{x}' \in B_i,\ i = 1, \ldots, K; \\ 0 & \text{otherwise.} \end{cases}$$

By rearranging observation indices such that the observations within a block are grouped together, we obtain a block-diagonal modulating matrix $\mathcal{K}_{block}(\mathscr{X}, \mathscr{X})$ with $\mathbf{1}_{n_i}\mathbf{1}_{n_i}^T$ the ith block, where $\mathbf{1}_{n_i}$ is a column vector of 1s and n_i is the number of observations within the ith block for $i = 1, 2, \ldots, K$. Thus the covariance matrix of the approximated residual process w_s on $\mathscr{X}$ is also block-diagonal, whose inverse can be efficiently computed if the block size is not large. We refer the FSA approach with $\mathcal{K} = \mathcal{K}_{block}$ as the *FSA-Block* method.

Let $\Sigma_{w^\dagger}$ denote the covariance matrix of observations $\mathscr{X}$ given by the FSA-Block method. It is positive definite (PD) when the knot set does not overlap with the observation set, otherwise it is positive semidefinite (PSD). To see why, note that $\Sigma_{w^\dagger} = \Sigma_{w_l} + \Sigma_{w_s}$, where Σ_{w_l} is the covariance matrix of the predictive process and Σ_{w_s} is a residual covariance. Here $\Sigma_{w_s} = (\Sigma_w - \Sigma_{w_l}) \circ \mathcal{K}_{block}(\mathscr{X}, \mathscr{X})$, where $\circ$ is the Schur product (entry-wise product) of matrices. Denote the observational locations in the block B_k by $\mathscr{X}_k$. Then we obtain a block diagonal matrix Σ_{w_s} with $\Sigma_{w_s}^k = \Gamma_w(\mathscr{X}_k, \mathscr{X}_k) - \Gamma_{w_l}(\mathscr{X}_k, \mathscr{X}_k)$ as its kth block. Since $\Sigma_{w_s}^k$ is the conditional covariance of $w(\mathscr{X}_k)$ given $w(\mathscr{X}^*)$, it is PD when $\mathscr{X}^* \cap \mathscr{X} = \emptyset$ and PSD otherwise. It follows that Σ_{w_s} and $\Sigma_{w^\dagger}$ are PD when $\mathscr{X}^* \cap \mathscr{X} = \emptyset$ and PSD otherwise.

The reduced-rank part plus the residual part using local partitioning provides an exact recovery of the true covariance within each subregion. Specifically, the covariance function of $w^\dagger$ is

$$\Gamma_{w^\dagger}(\mathbf{x},\mathbf{x}') = \begin{cases} \Gamma_w(\mathbf{x},\mathbf{x}') & \text{if } \mathbf{x},\mathbf{x}' \in B_i,\ i = 1,\ldots,K; \\ \Gamma_{w_l}(\mathbf{x},\mathbf{x}') & \text{otherwise.} \end{cases} \tag{3.5}$$

As the covariance approximation errors induced by the FSA-Block only occur for pairs belonging to different subregions and most of these pairs some distance apart, the errors $\Gamma_w(\mathbf{x},\mathbf{x}') - \Gamma_{w^\dagger}(\mathbf{x},\mathbf{x}')$ are expected to be small for most pairs.

In some cases, by taking advantage of the specific dependence structures, one can modify the general-purpose spatio-temporal FSA approach to achieve better covariance approximation and/or further reduce computational cost. For example, suppose $\Gamma_w(\mathbf{x},\mathbf{x}') = \Gamma_u(\mathbf{s},\mathbf{s}')\Gamma_v(t,t')$ where Γ_u and Γ_v are valid covariance functions in space and time domains, respectively, and that $n = N \times T$ observations are collected at spatial sites $\mathcal{S} = \{\mathbf{s}_1,\ldots,\mathbf{s}_N\}$ through time points set $\mathcal{T} = \{t_1,\cdots,t_T\}$. Then if we permute the observations by sorting the time in an increasing order, the covariance matrix of $w(\mathbf{s},t)$ at $\mathcal{S} \times \mathcal{T}$ can be written as $\Sigma_w = \Sigma_v \otimes \Sigma_u$, where $\Sigma_u = [\Gamma_u(\mathbf{s}_i,\mathbf{s}_j)]_{i,j=1:N}$, $\Sigma_v = [\Gamma_v(t_i,t_j)]_{i,j=1:T}$, and $\otimes$ is the Kronecker product. If $\epsilon(\mathbf{s},t) \equiv 0$, the resultant data covariance matrix $\Sigma_{\mathbb{Y}} = \Sigma_t \otimes \Sigma_{\mathbf{s}}$. The separability structure of $\Gamma_w(\mathbf{x},\mathbf{x}')$ can alleviate the computational demand because it reduces the dimension of the data covariance matrices that need to be inverted. Specifically, $|\Sigma_{\mathbb{Y}}| = |\Sigma_v|^T|\Sigma_u|^N$ and $\Sigma_{\mathbb{Y}}^{-1} = \Sigma_v^{-1} \otimes \Sigma_u^{-1}$. However, computational challenge still exists in the presence of large spatial locations and/or time points. In this case, the FSA approach can be applied to approximate the spatial covariance Σ_u and the temporal covariance Σ_v respectively.

3.2.3 Fast computation of parameter estimation and spatio-temporal prediction

In this section, we show the implementation of a spatio-temporal regression model using the FSA method. Replacing the latent spatio-temporal process w as (3.1) with its induced spatio-temporal FSA $w^{\dagger}$ as (3.2), we obtain the data model at n observed locations,

$$\mathbb{Y} = \mathbb{Z}\boldsymbol{\beta} + \mathbf{w}^{\dagger} + \epsilon, \qquad \epsilon \sim \text{MVN}(\mathbf{0}, \Sigma_{\boldsymbol{\epsilon}}), \tag{3.6}$$

where $\mathbf{w}^{\dagger}$ is an $n \times 1$ vector of $w^{\dagger}$ evaluated on $\mathscr{X}$. The data likelihood is then given by $\mathbb{Y} \sim \text{MVN}(\mathbb{Z}\boldsymbol{\beta}, \Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})$.

Here $\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}}$ is a sparse matrix for FSA-Taper or a block diagonal matrix for FSA-Block, whose inversion can be handled efficiently. We apply the Sherman-Woodbury-Morrison formula to calculate the inverse of $\Sigma_{\mathbb{Y}}$

$$\begin{aligned} \Sigma_{\mathbb{Y}}^{-1} &= (\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1} - (\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}\mathcal{C}(\mathscr{X}, \mathscr{X}^*) \\ &\quad \times \{\mathcal{C}^* + \mathcal{C}^T(\mathscr{X}, \mathscr{X}^*)(\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}\mathcal{C}(\mathscr{X}, \mathscr{X}^*)\}^{-1} \\ &\quad \times \mathcal{C}^T(\mathscr{X}, \mathscr{X}^*)(\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}. \end{aligned} \tag{3.7}$$

The determinant of $\Sigma_{\mathbb{Y}}$ can also be efficiently computed by applying Sylvester's determinant theorem,

$$\begin{aligned} |\Sigma_{\mathbb{Y}}| &= |\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}}| \times |\mathcal{C}^*|^{-1} \\ &\quad \times |\mathcal{C}^* + \mathcal{C}^T(\mathscr{X}, \mathscr{X}^*)(\Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}\mathcal{C}(\mathscr{X}, \mathscr{X}^*)|. \end{aligned} \tag{3.8}$$

Likelihood-based inference uses maximum likelihood or restricted maximum likelihood. For Bayesian inference, we need to specify priors for model parameters.

For the regression coefficient vector $\boldsymbol{\beta}$, we assign a vague multivariate normal prior $\boldsymbol{\beta} \sim \text{MVN}(\mu_{\boldsymbol{\beta}}, \Sigma_{\boldsymbol{\beta}})$. For the variance of measurement errors τ^2, we assign an inverse-gamma prior $\text{IG}(a, b)$ where the hyper-parameters a, b are chosen with reasonable guesses of mean and variance. Denote the set of parameters in the spatio-temporal covariance function Γ_w by $\boldsymbol{\theta}$, whose prior specification depends on the choice of the covariance function. Customarily, the inverse-gamma prior can be assigned on the variance parameter σ^2; the spatial/temporal range parameter can be assigned with a reasonably informative prior, e.g. a uniform prior with its support specified according to the belief on the practical spatial/temporal dependence range of the spatio-temporal dataset.

Let $\Omega = (\boldsymbol{\beta}, \boldsymbol{\theta}, \tau^2)$ be the collection of model parameters. The MCMC method is used to draw samples of parameters from the posterior

$$p(\Omega|\mathbb{Y}) \propto p(\boldsymbol{\beta})p(\boldsymbol{\theta})p(\tau^2)p(\mathbb{Y}|\Omega). \tag{3.9}$$

The Gibbs sampler is used to update $\boldsymbol{\beta}$ from $\text{MVN}(\boldsymbol{\mu}_{\boldsymbol{\beta}|\cdot}, \Sigma_{\boldsymbol{\beta}|\cdot})$, where

$$\begin{aligned} \Sigma_{\boldsymbol{\beta}|\cdot} &= (\mathbb{Z}^T(\Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}\mathbb{Z} + \Sigma_{\boldsymbol{\beta}}^{-1})^{-1}, \\ \boldsymbol{\mu}_{\boldsymbol{\beta}|\cdot} &= \Sigma_{\boldsymbol{\beta}|\cdot}(\mathbb{Z}^T(\Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}})^{-1}\mathbb{Y} + \Sigma_{\boldsymbol{\beta}}^{-1}\boldsymbol{\mu}_{\boldsymbol{\beta}}). \end{aligned}$$

For parameters without a closed-form of the full conditional distribution, we draw samples using the Metropolis-Hasting algorithm. For example, for spatial/temporal dependence range parameters, we can use truncated normal distribution centered at the current value as the proposal distribution. The log-normal proposal centered at the current value can also be used for dependence range parameters.

The spatio-temporal process regression model combined with the FSA provides a straightforward and efficient prediction using large spatio-temporal datasets. In classical geostatistics, assuming the model parameters are known, for a given new spatio-temporal point $\mathbf{x}_0$ the approximated best linear unbiased predictor (BLUP) of $Y(\mathbf{x}_0)$ is

$$\hat{Y}(\mathbf{x}_0) = Z^T(\mathbf{x}_0)\boldsymbol{\beta} + \mathcal{C}_{w^\dagger}(\mathbf{x}_0, \mathscr{X})\{\Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}}\}^{-1}(\mathbb{Y} - \mathbb{Z}\boldsymbol{\beta}) \tag{3.10}$$

and the approximated mean square error (MSE) is

$$\mathrm{MSE}(\hat{Y}(\mathbf{x}_0)) = \sigma^2 + \tau^2 - \mathcal{C}_{w^\dagger}(\mathbf{x}_0, \mathscr{X})\{\Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}}\}^{-1}\mathcal{C}^T_{w^\dagger}(\mathbf{x}_0, \mathscr{X}), \tag{3.11}$$

where $\mathcal{C}_{w^\dagger}(\mathbf{x}_0, \mathscr{X}) = [\Gamma_{w_l}(\mathbf{x}_0, \mathbf{x}_i) + \Gamma_{w_s}(\mathbf{x}_0, \mathbf{x}_i)]_{i=1:n, \mathbf{x}_i \in \mathscr{X}}$ is a $1 \times n$ cross-covariance matrix between $w^\dagger(\mathbf{x}_0)$ and $\mathbf{w}$. In practice, data-based estimates of the parameters are plugged in the above expression.

The Bayesian approach generalizes to the case of prediction when the covariance parameters are unknown. The predictive distribution for $Y(\mathbf{x}_0)$ is a Gaussian distribution with predictive mean given by (4.14) and variance given by (4.15). Therefore, a random sample of $Y(\mathbf{x}_0)$ from the (posterior) predictive distribution can be obtained by a draw of Ω from the posterior followed by a draw from the conditional predictive distribution of $Y(\mathbf{x}_0)$ given Ω.

Again, the calculation of BLUP and draws from the posterior predictive distribution involve the inversion of $\Sigma_{w_l} + \Sigma_{w_s} + \Sigma_{\boldsymbol{\epsilon}}$, which can be handled efficiently using the computational technique described in (3.7).

3.2.4 Selection of tuning parameters

Both the FSA-Taper and the FSA-Block involve tuning parameters: taper ranges and a knot set are required for the FSA-Taper; block partition in space-time domain and a knot set are required for the FSA-Block. The choices of these tuning parameters determine the approximation performance and the computational complexity of the FSA model.

Choice of knots is a key ingredient in the low rank component of the FSA. Typically, a denser knot design can lead to a better approximation of the parent process but at a cost of heavier computational burden. A heuristic way for selecting knots is to predetermine a knot number m based on available computational resources, then to place knots with good space-time coverage. Possible options include random sampling, Latin hypercube sampling [52, 65] and using a regular grid. Alternatively, one may consider a random knot selection in which knot number m and their locations are allowed to be chosen automatically.

For random knot selection, [34] introduced an adaptive predictive process model for spatial data. They fixed the knot number and modeled knot locations with a point pattern model. [41] applied the FSA-Taper approach to a nonstationary Matérn covariance function for spatial process, where the knot number was assigned with an improper flat prior on the set of all positive integers and knot locations were assigned with a uniform prior over the whole spatial domain.

Motivated by this work, we propose a Bayesian approach to adaptively select knot number and knot locations for the spatio-temporal FSA method. A RJMCMC algorithm [33] is offered to update the knot set from a discrete set of spatio-temporal points. Choices of candidate set include the set of all observed points or a regular grid covering the entire space-time domain, denoted by $\bar{\mathcal{L}}$. Let m be the knot number and $\mathcal{L}$ be the set of selected knot locations. We propose to assign the knot number m with a Poisson(λ) prior truncated at λ_0, where λ is chosen to balance the trade-off between computational capacity and model fitting, and $\lambda_0 > 0$ is set to reflect the maximum tolerance of computational time. Conditional on the knot number, we assume knots are randomly chosen from the candidate knot set, $p(\mathcal{L}|m) = \binom{M}{m}^{-1}$, where M is the size of $\bar{\mathcal{L}}$.

At each MCMC step, we consider three types of possible moves of selected knot set, changing from $(\mathcal{L}, m) \to (\mathcal{L}^*, m^*)$: (a) *birth*: add a knot by randomly selecting a point in $\bar{\mathcal{L}} \backslash \mathcal{L}$, so $m^* = m+1$; (b) *death*: randomly delete a knot in $\mathcal{L}$, so $m^* = m-1$; and (c) *change*: randomly choose a knot from $\mathcal{L}$ and then replace it with a randomly chosen point from $\bar{\mathcal{L}} \backslash \mathcal{L}$, so $m^* = m$. The acceptance ratio α of proposing a move is given by

$$\alpha = \min\left(1, \frac{p(\mathbb{Y}|\Omega, m^*, \mathcal{L}^*)p(\mathcal{L}^*|m^*)p(m^*)J((\mathcal{L}^*, m^*) \to (\mathcal{L}, m))}{p(\mathbb{Y}|\Omega, m, \mathcal{L})p(\mathcal{L}|m)p(m)J((\mathcal{L}, m) \to (\mathcal{L}^*, m^*))}\right).$$

Denote the probability of birth, death, and change moves with knot number m by b_m, d_m and c_m respectively, then $b_m + d_m + c_m = 1$. If $m = 1$, we set $d_m = c_m = 0$; if $m = \lambda_0$, we set $b_m = 0$ and $d_m = c_m = \frac{1}{2}$; and if $1 < m < \lambda_0$, we set $b_m = d_m = c_m = \frac{1}{3}$. Then J is calculated as follows,

$$J((\mathcal{L}, m) \to (\mathcal{L}^*, m^*)) = \begin{cases} \frac{b_m}{M-m} & \text{if } m^* = m+1, \\ \frac{d_m}{m} & \text{if } m^* = m-1, \\ \frac{c_m}{m(M-m)} & \text{if } m^* = m. \end{cases} \tag{3.12}$$

Following this RJMCMC algorithm, the knot number and locations are automatically selected at each iteration. We illustrate this algorithm in Section 3.2 through simulation experiments.

For the choice of block partition for the FSA-Block, one principle is to maximize residual correlations within blocks and minimize residual correlations across blocks so that most of the spatio-temporal correlations are preserved. If the spatio-temporal residual covariance is fairly isotropic, one simple strategy is to apply the K-means clustering algorithm on observed spatio-temporal points to find K cluster centers [44] and then create partitions in space-time domain. For the choice of tapering range for the FSA-Taper, some pilot studies can be conducted to give a rough estimate of the practical spatial/temporal dependence range. For example, we can select several time points and consider purely spatial datasets to estimate the spatial dependence range; similarly, we can consider time series at properly selected locations to estimate the time dependence range. These pilot estimates of dependence range are then subsequently used to set proper/conservative taper range to balance the trade-off between covariance approximation accuracy and computation efficiency.

3.2.5 Further improvement of computational efficiency by pre-tapering

Spatio-temporal datasets can be massive if they are observed at a large spatial domain during a long time period. Although the FSA approach is conceptually applicable to such datasets, direct application is impractical. A dense knot set is desirable to achieve a reasonable approximation, and the cross-covariance matrix. $\mathcal{C}(\mathscr{X}, \mathscr{X}^*)$ can bring challenges in matrix operations and storage. The computational issue is further complicated by a full MCMC implementation requiring a large number of iterations. Here we propose to pre-taper the original spatio-temporal covariance function, and then apply our FSA approach to the tapered covariance.

Consider a separable tapering function $\mathcal{K}(\mathbf{x}, \mathbf{x}') = \mathcal{K}_u(\mathbf{s}, \mathbf{s}'; \gamma_u)\mathcal{K}_v(t, t'; \gamma_v)$, where $\mathcal{K}_u$ and $\mathcal{K}_v$ are, respectively, spatial and temporal tapering functions. We create a sparse approximation to the original covariance Γ_w as

$$\Gamma_{\tilde{w}}(\mathbf{x}, \mathbf{x}') = \Gamma_w(\mathbf{x}, \mathbf{x}')\mathcal{K}(\mathbf{x}, \mathbf{x}').$$

This leads to a spatio-temporal covariance matrix sparser than the original covariance matrix. If the tapering is done conservatively, the tapered covariance is expected to retain most of the spatio-temporal dependence information. Since this matrix still keeps a large number of non-zero entries due to the massive size of the original covariance, we further reduce the computational cost by applying the FSA to $\Gamma_{\tilde{w}}(\mathbf{x}, \mathbf{x}')$.

Let $\Sigma_{\tilde{w}_l} + \Sigma_{\tilde{w}_s}$ be the FSA covariance matrix applied to the pre-tapered covariance matrix. Due to the pre-tapering, $\Sigma_{\tilde{w}_l}$ takes a quadratic form that involves a sparse $n \times m$ cross-covariance matrix $\mathcal{C}(\mathscr{X}, \mathscr{X}^*)$ and the inverse of a sparse $m \times m$ matrix $\mathcal{C}^*$. This greatly alleviates the computational burden in matrix operations and storage by applying sparse matrix techniques.

3.3 Simulation Studies

In this section, we report on simulation studies to evaluate the performance of the spatio-temporal FSA approach. In the first simulation study, we show the effectiveness of FSA-Block in approximating stationary spatio-temporal covariance models, and compare it with the independent blocks model (denoted as “Block”), predictive process model (denoted as “PP”) and modified predictive process model (denoted as “Modified PP”) ([21]). In the second simulation study, we illustrate the FSA with random knot selection for a nonstationary spatio-temporal covariance model. In both simulation studies, the full covariance model (denoted as “FM”) is also implemented to serve as the benchmark. The implementations of all methods were written in Matlab and run on a processor with 2.9 GHz Xeon CPUs and 16GB memory. For likelihood function optimization, we used the matlab function fminunc which implements a Broyden-Fletcher-Goldfarb-Shanno (BFGS) based Quasi-Newton method.

3.3.1 Simulation study 1

We randomly selected 4000 spatio-temporal locations on a space-time domain $\mathcal{S} \times \mathcal{T}$, where $\mathcal{S} = [0,\ 20] \times [0,\ 20]$ and $\mathcal{T} = [0,\ 20]$. The selected locations were then divided into a training set of size 3500 and a test set of size 500, where the test set included 243 points in a space-time hole $[5,\ 10] \times [5,\ 10] \times [0,\ 20]$ and 257 randomly selected points from the remaining space-time locations. We obtained realizations of the spatio-temporal process $Y(\mathbf{s}, t)$ at the selected points following the model in (3.1).

We first experimented with a nonseparable space-time covariance function proposed by [30],

$$C(\mathbf{h}, u) = \frac{\sigma^2}{\left(\frac{20|u|^{2\alpha}}{a} + 1\right)} \exp\left(-\frac{3\|\mathbf{h}\|}{c\left(\frac{20|u|^{2\alpha}}{a} + 1\right)^{\eta/2}}\right), (\mathbf{h}, u) \in \mathbb{R}^d \times \mathbb{R}, \tag{3.13}$$

where $a, c > 0$ are temporal and spatial dependence range parameters respectively; $\alpha \in (0, 1]$ is the smoothness parameter; and $\eta \in [0, 1]$ is the space-time interaction parameter. The mean of the regression model $\mu(\boldsymbol{s}, t)$ was set to 0 for the entire region. We used equal variance $\tau^2 = 0.01$ for the variance of $\epsilon(\mathbf{s}, t)$. The true values of the covariance parameters and other model parameters are shown in Table 3.1. Two parameter settings were considered: the first had $a = 10$ and $c = 20$, for a large-scale spatio-temporal dependence structure; the second had $a = 5$ and $c = 10$, for a small-scale spatio-temporal dependence structure. The maximum likelihood estimators (MLEs) were obtained based on the training set and the mean squared prediction errors (MSPE) were calculated based on the predictions of the test set for evaluation.

We implemented the FSA-Block approach using 500 spatio-temporal knots and 35 blocks. The knots were chosen randomly from $\mathcal{S} \times \mathcal{T}$ and the 35 block centers were created by the K-means clustering algorithm based on Euclidean distances of space-time points. For comparisons, the independent blocks method with the same 35 blocks and the predictive process method with the same 500 knots were considered.

The smoothness parameter α in the covariance model was fixed to be 0.5. The parameter estimations and the prediction results of different approaches are shown in the Table 3.1.

Table 3.1: The means and MSEs (in parenthesis) of each parameter and MSPE results for covariance model with a nugget. The results are based on 100 runs of simulations.

Method	Mean and MSEs					MSPE
	a	c	η	σ^2	τ^2	
	10	20	0.5	1	0.01	
FM	9.68 (1.75)	19.81 (4.31)	0.48 (0.0395)	0.97 (0.0062)	0.01 (0.0001)	0.34
FSA-Block	11.73 (5.84)	25.09 (34.71)	0.48 (0.0687)	1.04 (0.0104)	0.04 (0.0009)	0.37
Block	9.48 (2.38)	19.89 (4.97)	0.39 (0.0923)	0.96 (0.0090)	0.02 (0.0001)	0.43
PP	23.47 (206.36)	37.54 (349.67)	0.87 (0.2175)	2.25 (1.6948)	0.40 (0.1499)	0.45
Modified PP	26.16 (293.08)	42.26 (539.68)	0.86 (0.2224)	1.51 (0.2989)	0.03 (0.0011)	0.46
	5	10	0.5	1	0.01	
FM	5.10 (0.27)	10.19 (0.40)	0.47 (0.0368)	0.99 (0.0020)	0.02 (0.0004)	0.60
FSA-Block	5.82 (1.15)	11.68 (3.85)	0.46 (0.0809)	0.97 (0.0035)	0.06 (0.0031)	0.63
Block	5.03 (0.29)	10.18 (0.60)	0.43 (0.0560)	0.97 (0.0028)	0.03 (0.0006)	0.66
PP	17.75 (181.76)	21.24 (135.58)	0.62 (0.1947)	1.52 (0.3322)	0.64 (0.4006)	0.73
Modified PP	19.04 (221.35)	21.87 (150.45)	0.80 (0.2076)	1.18 (0.0554)	0.15 (0.0223)	0.73

Under the first parameter setting, where the spatio-temporal dependence range was large, the FSA-Block approach clearly outperformed the other methods in terms of prediction. The independent blocks method gave less accurate predictions. The predictive process model and the modified predictive process model did not work well either in terms of prediction, possibly requiring a denser knot set for a satisfactory approximation.

The FSA-Block obtained reasonable estimates for the range parameters a and c, but higher MSEs than the independent blocks method. The estimate of the nugget effect τ^2 obtained by the FSA-Block was slightly higher than the truth. The biases may be attributed to its predictive process part, which underestimates the correlations between blocks due to the limited number of knots. The naive independent blocks method worked well in terms of parameter estimation, which is not surprising since local information may be enough for estimating a stationary model with dense observations. The parameter estimation results of the predictive process model had noticeable biases, again perhaps due to the use of limited knots. Besides, the predictive process model gives a much larger estimate of the nugget effect due to its underestimation of the variance at each location [21]. The modified predictive process provided a bias correction for the variance at each location, so its estimates of τ^2 and σ^2 were better than those obtained from the predictive process model, but it still underestimated correlations, leading to biased estimation of range parameters. The FSA-Block provides bias-correction for the predictive process model within each block, thus the estimates of range parameters, the nugget, and the variance had much smaller biases than those obtained from the predictive process model and the modified predictive process model.

When the spatio-temporal dependence range was relatively small, the FSA-Block still gave comparable prediction performance with the full covariance model, while the predictive process model and the modified predictive process model gave worse prediction performance than under large-scale spatio-temporal dependence. The predictive process model fails to capture small-scale dependence and thus its performance is often sensitive to the strength of dependence, and its parameter estimation has fairly large biases. The FSA approach seems to be more robust and capable of adjusting the biases in the estimation of the range parameters at different scales of dependence range.

We compared the methods assuming no nugget effect in the covariance model, but do not include the results of the predictive process model because, without nugget it leads to a low-rank covariance matrix that can't be inverted. The results are presented in Table 3.2. Again, the FSA-Block outperforms the independent blocks method in terms of prediction. Besides, comparing with the case with the nugget effect (see Table 3.1), the FSA-Block provides more accurate parameter estimation when there is no nugget effect. In particular for set-up 2, the estimates of a, η, and σ^2 by the FSA-Block are very close to those from the full covariance model.

Table 3.2: The means and MSEs (in parenthesis) of each parameter and MSPE results for the covariance model without nugget. The results are based on 100 runs of simulations.

Settings	Method	Mean and MSEs				MSPE
		a	c	η	σ^2	
Set-up 1		10	20	0.5	1	
	FM	10.16 (1.14)	20.32 (2.43)	0.53 (0.0373)	1.01 (0.0048)	0.33
	FSA-Block	10.52 (1.58)	22.55 (10.03)	0.46 (0.0433)	1.05 (0.0076)	0.37
	Block	9.76 (1.32)	19.81 (1.95)	0.48 (0.0649)	0.99 (0.0042)	0.42
Set-up 2		5	10	0.5	1	
	FM	5.01 (0.27)	10.01 (0.34)	0.51 (0.0311)	1.00 (0.0021)	0.59
	FSA-Block	5.11 (0.28)	10.67 (0.94)	0.53 (0.0379)	1.01 (0.0021)	0.62
	Block	4.90 (0.28)	9.98 (0.47)	0.47 (0.0501)	0.99 (0.0024)	0.66

The Matérn class [51, 62] is another widely used stationary covariance family due to its flexibility in accommodating different smoothness. We simulated data from the Matérn covariance model with

$$C(\mathbf{h}, u) = \frac{\sigma^2}{\Gamma(\nu)2^{\nu-1}} \left(3\sqrt{\frac{\|\mathbf{h}\|^2}{\phi_s^2} + \frac{|u|^2}{\phi_t^2}} \right)^{\nu} K_\nu \left(3\sqrt{\frac{\|\mathbf{h}\|^2}{\phi_s^2} + \frac{|u|^2}{\phi_t^2}} \right), \tag{3.14}$$

where $\phi_s, \phi_t > 0$ are spatial and temporal range parameters, respectively, $\nu > 0$ is the smoothness parameter, and K_ν denotes the modified Bessel function of the second kind of order ν. We experimented with $\nu = 0.5, 1, 2$ to investigate the performance of the FSA-Block method in terms of parameter estimations and predictions under different level of smoothness. Other particulars were the same as in the simulation study for Gneiting's nonseparable covariance model. The MLEs were obtained for all model parameters and we took $\nu \in (0, 3]$ when estimating it. The parameter estimations and prediction results are shown in Table 3.3.

Table 3.3: The means and MSEs (in parenthesis) of each parameter and MSPE results for Matérn's covariance model. The results are based on 100 runs of simulations.

Method	Mean and MSEs					MSPE
	ϕ_t	ϕ_s	ν	σ^2	τ^2	
	5	10	0.5	1	0.01	
FM	4.65 (0.68)	9.23 (2.45)	0.54 (0.0041)	0.96 (0.0102)	0.02 $(4.1 \cdot 10^{-4})$	0.37
FSA-Block	4.42 (0.86)	8.87 (3.24)	0.61 (0.0196)	0.95 (0.0113)	0.05 $(2.0 \cdot 10^{-3})$	0.39
Block	4.46 (0.73)	8.87 (2.81)	0.55 (0.0044)	0.94 (0.0107)	0.02 $(3.8 \cdot 10^{-4})$	0.42
PP	2.13 (8.27)	4.19 (33.89)	3.00 (6.2494)	1.13 (0.0408)	0.34 $(1.1 \cdot 10^{-1})$	0.46
Modified PP	2.15 (8.17)	4.24 (33.30)	3.00 (6.2499)	0.90 (0.0235)	0.25 $(5.7 \cdot 10^{-2})$	0.46
	5	10	1	1	0.01	
FM	4.90 (0.54)	9.71 (1.99)	1.02 (0.0051)	0.97 (0.0213)	0.01 $(6.6 \cdot 10^{-6})$	0.11
FSA-Block	4.73 (0.59)	9.44 (2.21)	1.08 (0.0134)	0.98 (0.0217)	0.01 $(2.0 \cdot 10^{-5})$	0.13
Block	4.69 (0.57)	9.31 (2.25)	1.04 (0.0068)	0.94 (0.0245)	0.01 $(7.3 \cdot 10^{-6})$	0.16
PP	2.69 (5.43)	5.33 (21.99)	2.98 (3.9449)	1.78 (0.7571)	0.11 $(1.1 \cdot 10^{-2})$	0.18
Modified PP	2.73 (5.24)	5.42 (21.19)	2.99 (3.9710)	1.19 (0.1121)	0.07 $(3.3 \cdot 10^{-3})$	0.18
	5	10	2	1	0.01	
FM	4.83 (0.39)	9.61 (1.54)	2.06 (0.0281)	0.95 (0.0422)	0.01 $(2.2 \cdot 10^{-7})$	0.02
FSA-Block	4.81 (0.56)	9.56 (2.04)	2.21 (0.0963)	1.13 (0.1020)	0.01 $(4.7 \cdot 10^{-7})$	0.02
Block	4.56 (0.66)	9.06 (2.64)	2.15 (0.0709)	0.90 (0.0551)	0.01 $(2.7 \cdot 10^{-7})$	0.13
PP	4.61 (0.34)	9.14 (1.21)	2.60 (0.3807)	2.23 (1.6538)	0.02 $(1.1 \cdot 10^{-4})$	0.03
Modified PP	4.81 (0.23)	9.59 (0.76)	2.56 (0.3500)	1.42 (0.2555)	0.01 $(2.6 \cdot 10^{-6})$	0.03

Overall, the FSA-Block method and independent blocks method give reasonable estimates of the model parameters for different true values of ν, while the predictive process/modified predictive process tends to overestimate ν and gives notably biased estimates of other parameters, especially when ν is relatively small. In terms of prediction performance, it appears that the FSA-Block method achieves comparable results to the full covariance model under all the three parameter settings. The prediction performance of the other three competing methods depends on the true value of smoothness parameter ν. Specifically, the independent blocks method achieves good prediction results and outperforms the predictive process/modified predictive process models when ν is relatively small, but its prediction performance is inferior to the predictive process/modified predictive process models when ν is large.

3.3.2 Simulation study 2

We applied the RJMCMC algorithm described in Section 2.4 to automatically select knots when applying the FSA. We used the same set of space-time locations as in simulation study 1 and we simulated $Y(\mathbf{s}, t)$ at these locations following the model in (3.1), with $\mu = 0$ and $\tau^2 = 0$. The spatio-temporal random effect w was assumed to have a separable nonstationary correlation function $\Gamma_w(\mathbf{s}_i, t_i; \mathbf{s}_j, t_j) = \Gamma_u(\mathbf{s}_i, \mathbf{s}_j) \cdot \Gamma_v(t_i, t_j)$, where Γ_u and Γ_v were nonstationary spatial and temporal covariances constructed following [54]. Here the square spatial domain $[0,\ 20] \times [0,\ 20]$ was divided equally into 4 subregions with $D(\mathbf{s}_i)$ such that $D(\mathbf{s}_i) = l$ if $\mathbf{s}_i$ belong to the lth subregion,

$$\Gamma_u(\mathbf{s}_i, \mathbf{s}_j) = |\mathbf{H}_{D(\mathbf{s}_i)}|^{\frac{1}{4}} |\mathbf{H}_{D(\mathbf{s}_j)}|^{\frac{1}{4}} \left| \frac{\mathbf{H}_{D(\mathbf{s}_i)} + \mathbf{H}_{D(\mathbf{s}_j)}}{2} \right|^{-\frac{1}{2}} \exp(\sqrt{Q_{ij}}),$$

where $Q_{ij} = (\mathbf{s}_i - \mathbf{s}_j)^T \left(\frac{\mathbf{H}_{D(\mathbf{s}_i)} + \mathbf{H}_{D(\mathbf{s}_j)}}{2} \right)^{-1} (\mathbf{s}_i - \mathbf{s}_j)$ is the Mahalanobis distance. $\mathbf{H}_{D(\mathbf{s})}$ is referred to as the kernel covariance matrix. The eigenvalue decomposition of $\mathbf{H}_{D(\mathbf{s})}$ has clear geometric interpretations: the square roots of the eigenvalues of $\mathbf{H}_{D(\mathbf{s})}$ control the range of the spatial dependence and the eigenvector matrix corresponds to a rotation matrix. We reparameterize $\mathbf{H}_{D(\mathbf{s})}$ as

$$\mathbf{H}_{D(\mathbf{s})} = \mathbf{R}(\theta_{D(\mathbf{s})}) \begin{pmatrix} \lambda_{D(\mathbf{s}),1} & 0 \\ 0 & \lambda_{D(\mathbf{s}),2} \end{pmatrix} \mathbf{R}^T(\theta_{D(\mathbf{s})}),$$

where $\lambda_{D(\mathbf{s}),1}, \lambda_{D(\mathbf{s}),2}$ are eigenvalues of $\mathbf{H}_{D(\mathbf{s})}$, and $\mathbf{R}(\theta_{D(\mathbf{s})})$ is a rotation matrix. Anisotropy is introduced to the covariance function by allowing different values of $\lambda_{D(\mathbf{s}),1}$ and $\lambda_{D(\mathbf{s}),2}$. Nonstationarity is achieved by assuming a spatially-varying kernel covariance across different subregions. The time domain was divided equally into 2 intervals and Γ_v was constructed in a similar way as Γ_u. In this experiment, for simplicity, we set $\mathbf{R}$'s to be identity matrices. We used the same 35 blocks as in simulation study 1 for the FSA-Block approach. The true values of model parameters are shown in the second column of Table 3.4.

For Bayesian posterior inferences, flat priors were adopted for all λ's, and truncated normal distributions on $(0,\ 50)$ were used as their proposal distributions. The prior for knot number m was set to be Poisson(50), truncated at $\lambda_0 = 700$. Conditional on the knot number, we assign uniform priors from the set containing all observed space-time points for knot locations. We then followed the RJMCMC algorithm in Section 2.4 to draw samples of knots. We compared this method with the FSA-Block approach using the fixed knot design, where the knot set was predetermined by choosing a random sample from the observed location set. We ran 7000 iterations after a burning period of 1000 iterations. 3500 posterior samples were collected with thinning, using every 3rd iteration.

In Table 3.4, the fixed and random knot designs give fairly close estimates of the covariance parameters. These estimates are also close to those from the full model, suggesting that the FSA-Block is capable of providing a good approximation to a

Table 3.4: Parameter estimation and prediction results for FSA-Block approach with knots selected by RJMCMC algorithm. For number of knots m, we report its posterior means.

Parameters	True value	Full Model	Random knots	$m = 50$	$m = 100$	$m = 300$
$\lambda_{\mathbf{s}_{11}}, \lambda_{\mathbf{s}_{12}}$	40	39.81 (4.28)	42.95 (3.79)	42.22 (4.10)	39.45 (4.37)	40.86 (4.10)
$\lambda_{\mathbf{s}_{21}}, \lambda_{\mathbf{s}_{22}}$	25	20.91 (2.53)	22.75 (2.70)	19.76 (2.32)	20.86 (2.38)	21.95 (2.48)
$\lambda_{\mathbf{s}_{31}}, \lambda_{\mathbf{s}_{32}}$	20	23.37 (2.82)	23.93 (2.95)	24.64 (3.23)	22.97 (2.92)	22.27 (2.84)
$\lambda_{\mathbf{s}_{41}}, \lambda_{\mathbf{s}_{42}}$	10	10.08 (1.07)	10.13 (1.14)	9.98 (1.11)	9.87 (1.18)	9.42 (1.07)
λ_{t_1}	40	37.06 (4.98)	43.75 (4.28)	41.93 (4.80)	41.52 (4.92)	40.40 (5.03)
λ_{t_2}	10	11.59 (1.53)	12.23 (1.64)	11.86 (1.65)	11.51 (1.50)	11.81 (1.64)
MSPE	-	0.231(0.001)	0.265(0.008)	0.287(0.001)	0.271 (0.001)	0.262(0.001)
m	-	-	82.90	-	-	-
Time(hour)	-	25.49	8.05	6.35	7.75	8.90

nonstationary spatio-temporal covariance model. The prediction performance of the FSA-Block approach with the fixed knot design seems to depend on the knot number. Under the random knot scenario, the posterior mean of the knot number m given by the RJMCMC algorithm is close to 83, but its MSPE is just slightly larger than that of the FSA with 300 fixed knots, indicating that the RJMCMC algorithm can be effective in determining a reasonable knot number and selecting the "most useful" knots from a candidate set.

3.4 Analysis of The Eastern US Ozone Data

We applied the spatio-temporal FSA to the daily surface ozone data collected at 513 monitoring stations in the eastern US from May 1, 1998 to October 31, 1999. The observations are the maxima of hourly means over 8 consecutive hours of ozone. The raw data can be downloaded from *www.image.ucar.edu/Data/Ozmax/*.

We followed the procedure described in [28] and [6] to pre-process the daily observations. The daily maximum 8-hour ozone measurement at station $\mathbf{s}$ and day t is assumed to have the decomposition,

$$Y(\mathbf{s},t) = \mu(\mathbf{s},t) + \sigma(\mathbf{s})w(\mathbf{s},t),$$

where $\mu(\mathbf{s},t) = a(\mathbf{s}) + \sum_{j=1}^{3}\{b_j(\mathbf{s})\cos(2\pi jt/184) + c_j(\mathbf{s})\sin(2\pi jt/184)\}$, modeling the seasonal effect. The coefficients in the seasonal effect $\mu(\mathbf{s},t)$ were estimated by ordinary least square and $\sigma(\mathbf{s})$ was estimated using the residuals after removing the seasonal effect. Following [28], the estimated coefficients matrix of the seasonal effect were further smoothed over space. Figure 3.1 shows the locations of the 513 monitoring stations and the seasonal effect in 1999.

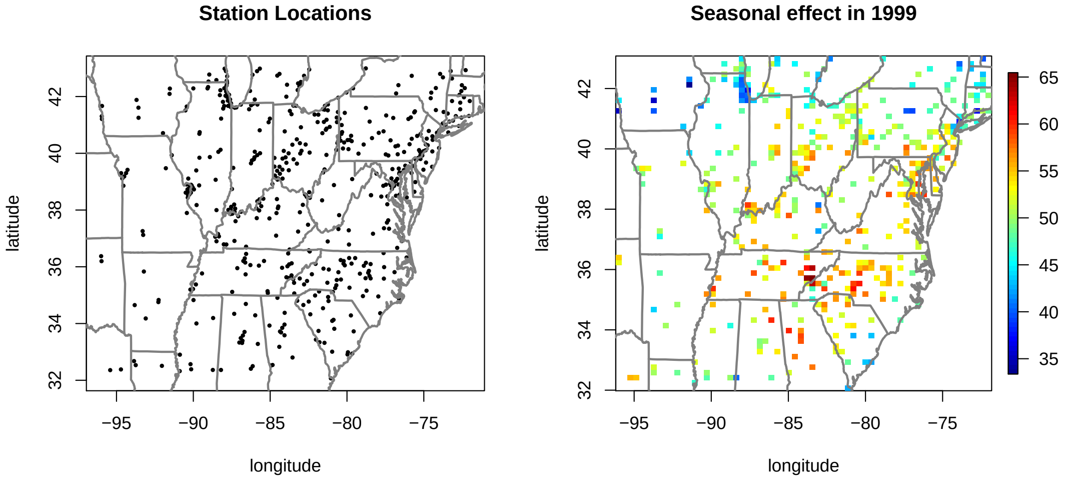


Figure 3.1: 513 ozone monitoring station locations and the estimated averaged seasonal effect in 1999.

We modeled the spatio-temporal component $w(\mathbf{s},t)$ by a Gaussian process with mean 0 and a nonseparable spatio-temporal covariance function as in (3.13), with $\mathbf{s}$ defined on the sphere. Since the station locations are on the sphere, the transformed great circle distance [29, 28] was used to ensure positive-definiteness of the covariance function: $d(\mathbf{s}, \mathbf{s}') = 2r \sin\left(\frac{\Delta\phi}{2}\right)$, where r is the radius of the earth and $\Delta\phi \in [0,\ \pi]$ is the central angle between $\mathbf{s}$ and $\mathbf{s}'$. We used kilometer as the unit of spatial lags and day as the unit of temporal lags.

Using only the monthly data in June and July, 1998 and 1999, allows us to implement the full covariance model whose results can be used as a benchmark. For each monthly dataset containing roughly $15,000$ observations, we randomly selected 1500 space-time data points as a hold-out set for prediction and used the rest as training data. We obtained the maximum likelihood estimates of model parameters of the full model, the FSA-Block method, and the weighted composite likelihood (WCL) method [6] for the training data. For the FSA-Block approach, we applied Latin hypercube sampling to obtain 400 space-time knots and the K-means clustering algorithm to divide the monthly data into 14 blocks. The WCL method needs to specify a pair of spatio-temporal lags (d_s, d_t) such that the weights $w_{ij} = 1$ when $d(\mathbf{s}, \mathbf{s}') \leq d_s$ and $|t_i - t_j| \leq d_t$, $w_{ij} = 0$ otherwise. Following [6], we set $d_s = 400, d_t = 3$, obtained by minimizing the asymptotic variances of WCL estimators.

Table 3.5 shows the parameter estimation and prediction results for each monthly dataset in June and July in 1998 and 1999. In general, parameter estimates of the FSA-Block method are close to those from the full model, implying that the FSA-Block method can approximate the original model reasonably well. The WCL method overestimates the spatial range parameter c. This may be due to the fact that it only includes pairs within certain distance for inference and thus may fail to incorporate large-scale dependence information. As the WCL estimate of η is always on the boundary of its parameter space, the estimate of α has the same problem in half of the cases, indicating possible convergence problems when using the WCL for parameter estimation.

We focused on comparing the prediction performance using the parameter estimates from the WCL, the FSA-Block, and the full model. Since the training set is large, computation of the BLUP using all training data is expensive, we used the training data within 4 days from each test data point when computing the BLUP. Although our FSA-Block method can be applied to the full training set and provides efficient computation for the BLUP, we used the partial training data and the full covariance model for fair comparison. Table 3.5 shows that the prediction results of the FSA-Block method and the full model are close to each other, and that both methods provide slightly better prediction performance than the WCL method.

Table 3.5: Parameter estimation and prediction results of monthly data. The root mean squared predictive errors (RMSPE) were made based on the within 4 days' training data.

Month	Methods	a	c	η	α	RMSPE
1998, June	FSA	22.70	414.40	0.023	0.212	0.335
	WCL	24.73	1299.63	1	0.503	0.339
	FullModel	20.95	414.22	0.033	0.149	0.334
1998, July	FSA	21.94	358.58	0.092	0.167	0.359
	WCL	25.77	1119.10	1	0.3467	0.359
	FullModel	21.28	356.99	0.076	0.138	0.359
1999, June	FSA	20.63	454.71	0.061	0.305	0.317
	WCL	17.69	1027.91	1	1	0.330
	FullModel	18.45	467.64	0.042	0.233	0.317
1999, July	FSA	23.91	338.94	0.036	0.260	0.430
	WCL	16.90	827.70	1	1	0.450
	FullModel	20.36	353.93	0.010	0.181	0.430

We considered larger datasets of around 45000 daily ozone observations from June to August in 1998 and 1999. We randomly held out 4500 space-time data points for prediction. We considered three covariance models to fit the data: model A is the separable covariance model in (3.13) with $\eta = 0$; model B is the nonseparable covariance model in (3.13); and model C is the Matérn covariance model in (3.14).

Here MLEs of model parameters were only obtained for the FSA-Block and WCL methods only since the full covariance model is not computationally feasible. For the WCL method, the weights were chosen in the same way as in the monthly data analysis; for the FSA-Block method, we applied Latin hypercube sampling to obtain 400 space-time knots and the K-means clustering algorithm to divide each summer dataset into 54 blocks. Prediction was performed for both methods using partial training data.

Parameter estimations and prediction results are shown in Table 3.6. It appears that Gneiting's covariance models (Model A and B) outperform the Matérn covariance model (Model C) in terms of prediction. The separable and the nonseparable Gneiting models provide comparable prediction results, indicating that a simple separable covariance model may be capable of modeling the spatio-temporal dependence of the summer ozone datasets. For all three covariance models, the FSA-Block clearly outperforms the WCL method in terms of prediction performance. The parameter estimations using WCL seem problematic in some cases, for example, estimates of η and α for Gneiting models are on the boundary of the parameter space for the dataset in the summer of 1999.

We applied the RJMCMC algorithm described in Section 3.2.4 to automatically select knots when applying the FSA-Block on the summer ozone datasets in 1998 and 1999. We considered only Gneiting's model (3.13) since it achieves better prediction performance than the Matérn covariance model. As the MLEs of η are close to zero for both datasets, and there is no significant difference between the separable and nonseparable models in terms of the parameter estimates of other parameters and prediction, we only applied model A here. For Bayesian inference, a uniform prior with support specified according to the prior belief on the practical range was assigned to the spatial/temproal range parameter. A uniform prior on $(0, 1]$ was

Table 3.6: Parameter estimation and prediction results of summer ozone. Model A is the separable covariance model in (3.13) with space-time interaction parameter $\eta = 0$. Model B is the nonseparable covariance model in (3.13). And model C is the Matérn covariance model in (3.14).

		Gneiting's model					
Year	Method	Model	a	c	η	α	RMSPE
1998	FSA	A	20.17	376.88	–	0.268	0.372
		B	20.78	378.62	0.062	0.267	0.372
	WCL	A	29.22	1106.17	–	0.716	0.385
		B	23.98	1060.31	1	0.775	0.382
		Matérn model					
		Model		ϕ_t	ϕ_s	ν	RMSPE
	FSA	C		2.55	1958.93	0.274	0.436
	WCL	C		12.30	285.89	1.830	0.635
		Gneiting's model					
Year	Method	Model	a	c	η	α	RMSPE
1999	FSA	A	19.15	418.08	–	0.270	0.380
		B	19.22	418.32	0.008	0.270	0.380
	WCL	A	20.01	1022.59	–	1	0.402
		B	14.63	1001.88	1	1	0.401
		Matérn model					
		Model		ϕ_t	ϕ_s	ν	RMSPE
	FSA	C		1.49	2863.92	0.251	0.447
	WCL	C		9.91	132.79	2.750	0.821

taken for α, and a Poisson(100) prior truncated at $\lambda_0 = 700$ was assigned to the knot number m. We ran 8000 iterations to collect 2000 posterior samples after a burn-in period of 4000 iterations, thinning by using every third iteration. The posterior prediction was calculated using the partial training data by plugging in the maximum a posteriori (MAP) estimates of the knot set and the other model parameters. We used the same number of blocks as in the MLE cases.

Table 3.7 shows the Bayesian posterior sample means and standard deviations (in the parenthesis) of model parameters from FSA-Block with random knot selection for the summer ozone datasets in 1998 and 1999. For both datasets, the posterior sample means of range parameters are close to the corresponding MLE estimates

Table 3.7: Parameter estimation and prediction results of summer ozone datasets by FSA-Block with random knot selection.

Method	Year	a	c	α	m	RMSPE
FSA	1998	18.80(0.46)	380.82(4.05)	0.164(0.009)	247.74	0.373
	1999	18.83(0.40)	417.70(3.90)	0.191(0.009)	246.92	0.380

with 400 fixed knots, while the posterior sample mean of α is slightly less than its MLE counterpart. For each summer ozone dataset, the posterior mean of the knot number m is about 250, but its RMSPE is about the same as that from the MLE result with 400 fixed knots (see Table 3.6), implying that the RJMCMC algorithm can be effective in determining a reasonable knot number and selecting useful knots from the candidate set.

3.5 Discussion

We have proposed a method FSA to approximate a spatio-temporal covariance function. Our construction provides a flexible framework for statistically and computationally efficient parameter estimation and prediction for modeling of large spatio-temporal datasets. We have focused on the FSA-Block variation that provides exact bias-corrections for spatio-temporal pairs of the covariance matrix within blocks.

4. APPLICATIONS OF GAUSSIAN PROCESS MODEL TO UNCERTAINTY QUANTIFICATION OF COMPUTER CODE OUTPUTS

4.1 Introduction

The computer model plays a crucial role in scientific research for studying behaviors of complex systems through computer experiments. In the context of Uncertainty Quantification (UQ), a key question of interest is to examine how computer model outputs change with different configurations of input parameters controlling physical variables, initial or boundary conditions, and so on. Although a computer model with a fine resolution is desired since it often produces more accurate simulations, it can be computationally prohibitive to produce a large number of fine resolution simulation runs at different input values. This motivates the use of computationally inexpensive surrogate models to facilitate learning of response surface.

Gaussian process models were first used in [17] and [58] for building surrogate models for computer experiments. Oakley and O'Hagan [53] later applied Gaussian process emulators for uncertainty quantification under the Bayesian framework. Covariance function is a key ingredient in such models since it determines the dependence structure of the Gaussian process. In the context of Gaussian process emulators, the most widely used auto-covariance function is usually stationary and separable in each input dimension; the cross-covariance among outputs is also assumed to be separable from dependence in other dimensions for mathematical tractability. For example, [10] proposed a stationary multi-output Gaussian process emulator based on separable cross-covariance. Also based on separable cross-covariance, [46] generalized the work in [10] and [32] to a Bayesian Treed multivariate Gaussian process model, accounting for both the nonstationarity and the multivariate features of the data.

The assumption of separability allows fitting Gaussian process model in each input dimension separately. It leads to a separable covariance structure of covariance function and hence alleviate the computational demand by reducing the dimension of the covariance matrices to be inverted. One such example is in [7], where they introduced a multi-output separable Gaussian process model assuming the auto-covariance function of each output is separable in input, space and time. Then by making use of the properties of Kronecker product, the inverse of the covariance matrix of one output can be decomposed into the Kronecker product of inverses of an input covariance, a purely spatial covariance, and a purely temporal covariance, all of which typically have reduced dimensions so that data likelihood can be evaluated efficiently. Although the separable auto-covariance model has the aforementioned merits, it suffers from several limitations. First, it is lack of flexibility to allow for interactions between different types of correlations. [12] pointed out that if a stationary spatio-temporal covariance function is separable, then the temporal dependence structure can not vary spatially and the spatial dependence structure can not vary temporally. However, in spatio-temporal statistics, the space-time interaction effect is often of particular interest. Such a limitation is also encountered by the separable emulator; the dependence structure of one input dimension is not allowed to change with other input dimensions. Second, the separable covariance function also has implications on conditional independence of outputs [45]. For instance, given a stationary bivariate Gaussian process $f(\cdot, \cdot)$ with a separable covariance function, it can be shown that $f(\xi, t)$ and $f(\xi', t')$ are independent given $f(\xi, t')$. A more comprehensive discussion of separable model can be found in [55].

Since the separable covariance may be restrictive in some cases, it is often desirable to consider a more general class of nonseparable auto-covariance models. In spatio-temporal statistics, much work have been done to construct flexible classes of nonseparable auto-covariance functions in space and time [12, 30, 66]. Typically the nonseparable space-time model has a parameter $\beta \in [0, 1]$, referred to as the spatio-temporal interaction parameter, and the model reduces to be separable when $\beta = 0$. More sophisticated nonseparable covariance model of three or higher input dimensions can be constructed following the work by [1], where they extended methods in [30] to propose a nonseparable cross-covariance model for multivariate random fields. Motivated by these work in spatial statistics, we develop a flexible class of nonseparable auto-covariances for uncertainty quantification of computer models. In particular, this class of models includes separable models as special cases.

For computations, it is well known that the Gaussian process model scales badly with sample size n, requiring $\mathcal{O}(n^3)$ order of computations. Large sample size n typically makes the computations for the Gaussian process emulator prohibitive, unless some particular structures of the covariance functions are assumed, e.g. the separability. To overcome the computational bottleneck, we introduce the Full-Scale approximation (FSA) approach to reduce computations [59, 60], which applies to both separable and nonseparable covariance structure. The FSA approach combines the ideas of a reduced rank Gaussian process [4] and covariance tapering [43] to provide a satisfactory approximation of the original covariance, under both large and small dependence scales of the data. Its computational complexity is linear with n, reducing the computational cost significantly.

The major contributions of this Section have two folds: first we propose a flexible class of nonseparable auto-covariance functions for each computer output to model the interaction effect among input, space and time. This class of models relaxes the separability assumption that is typically made for Gaussian process emulators and provides a more flexible and general tool to describe dependence for computer model outputs. Second, we introduce the FSA approach to provide efficient computations for nonseparable Gaussian process emulator. Since the FSA approach applies to any given covariance structure of a computer model output, it can also be combined with separable model to further reduce computational cost in the case when certain input dimensions have large sample sizes for simulation accuracy. In this paper, we illustrate our method assuming a stationary covariance function for each computer model output. We remark that our computational approach directly applies to nonstationary covariance functions as well.

4.2 Methodology

We consider a physical problem with input domain $\mathcal{X}_\xi \subseteq \mathbb{R}^{k_\xi}$, spatial domain $\mathcal{X}_s \subseteq \mathbb{R}^{k_s}$ and temporal domain in an interval $\mathcal{X}_t = [0, T] \subseteq R^+$, where k_ξ, k_s are the dimensions of the input and spatial domain. The input domain $\mathcal{X}_\xi$ is usually assumed to bounded and can thus be considered as a compact subset of $\mathbb{R}^{k_\xi}$.

In computer simulations, spatial and temporal domain are often fixed while simulations are run at a set of samples from the input domain. Therefore, we can represent the whole domain as a tensor product of the input, spatial, and temporal domain. For an input parameter $\boldsymbol{\xi} \in \mathcal{X}_\xi$, the computer simulation returns the (multi-output) response on a given (a priori known) set of n_s spatial points $(\mathbf{s}_1, \dots, \mathbf{s}_{n_s})^T$ and n_t time steps $(t_1, \dots, t_{n_t})^T$. A single choice from the input domain $\boldsymbol{\xi}$ generates a multi-output response data which can be represented as a $(n_s n_t) \times q$ matrix, where q is the number of the output variables from a computer simulation.

Let $n = n_\xi n_s n_t$ denote the total sample size. We define $\mathbf{x}_i = (\boldsymbol{\xi}_i, \mathbf{s}_i, t_i)$, denoting an input, space and time point for $i = 1, \cdots, n$. For modeling reasons we represent the output as a q multivariate response $\mathbf{f}(\mathbf{x}_i) = \mathbf{f}(\boldsymbol{\xi}_i, \mathbf{s}_i, t_i) \in \mathbb{R}^q$. For simplicity, we call the input domain, spatial domain, and temporal domain as input, space, and time respectively throughout this paper. We will collectively denote input of $\mathbf{f}(\cdot)$ by $\mathbf{x} = (\boldsymbol{\xi}, \mathbf{s}, t)$ and the domain of $\mathbf{x}$ by $\mathscr{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$.

4.2.1 Multivariate Gaussian process regression model

We model $\mathbf{f}(\mathbf{x})$ as a q-dimensional Gaussian process:

$$\mathbf{f}(\cdot)|B, \tilde{\boldsymbol{\theta}} \sim \mathcal{N}_q(\boldsymbol{\mu}(\cdot; B), \Gamma(\cdot, \cdot; \tilde{\boldsymbol{\theta}})), \tag{4.1}$$

where $\boldsymbol{\mu}(\cdot; B)$ is the mean function and $\Gamma(\cdot, \cdot; \tilde{\boldsymbol{\theta}})$ is the covariance function of the q-dimensional Gaussian process $\mathbf{f}(\mathbf{x})$. A typical choice of the mean function $\boldsymbol{\mu}(\cdot; B)$ is the linear regression model: $\boldsymbol{\mu}(\mathbf{x}) = \mathbf{h}^T(\mathbf{x})B$, where $\mathbf{h}(\mathbf{x})$ is formed by m basis functions evaluated at $\mathbf{x}$ and B is a $m \times q$ unknown regression coefficients matrix. The cross-covariance is often assumed to be separable from other dependence [50], that is, $\Gamma(\cdot, \cdot; \tilde{\boldsymbol{\theta}}) = \rho(\cdot, \cdot; \boldsymbol{\theta})\Sigma$, where Σ is the covariance matrix that models the cross-dependence structure of q distinct components of $\mathbf{f}(\cdot)$, and $\rho(\cdot, \cdot; \boldsymbol{\theta})$ is the auto-correlation within each component. Let the correlation parameter vector $\boldsymbol{\theta} \in \Theta$ and have dimension d_θ. In this work we will follow the assumption that the cross-covariance among multivariate components and auto-correlation within each component are separable for simplicity. More general nonseparable cross-covariance models can be constructed following the work in [22].

Let $Y = (\mathbf{f}^T(\mathbf{x}_1), \mathbf{f}^T(\mathbf{x}_2), \cdots, \mathbf{f}^T(\mathbf{x}_n))^T$ be the $n \times q$ matrix of computer model output, $H = (\mathbf{h}(\mathbf{x}_1), \ldots, \mathbf{h}(\mathbf{x}_n))^T \in \mathbb{R}^{n \times m}$ be the design matrix in the regression mean function, and $R = [\rho(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1:n} \in \mathbb{R}^{n \times n}$ be the correlation matrix at a given set of n points $\mathscr{X}$. The data likelihood function is given by the following matrix normal distribution

$$\begin{aligned} Y|B,\Sigma,\boldsymbol{\theta} &\sim \mathcal{N}_{n\times q}(HB,R,\Sigma), \qquad (4.2)\\ L(Y|B,\Sigma,\boldsymbol{\theta}) &\propto |R|^{-q/2}|\Sigma|^{-n/2}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\Sigma^{-1}(Y-HB)^T R^{-1}(Y-HB)\right)\right). \end{aligned}$$

To evaluate the above likelihood function, we need to compute the determinant and inverse of the $n\times n$ matrix R. When n is very large, the computation burden can often lead to failures in calculating these quantities. In Section 2.3, we will introduce a covariance approximation method to facilitate the computations for likelihood evaluations.

Squared exponential kernel function is one classical choice of the separable autocorrelation function $\rho(\cdot,\cdot)$,

$$\rho(\mathbf{x},\mathbf{x}') = \exp\left(-\sum_{i=1}^{k_\xi}\frac{(\xi_i-\xi_i')^2}{\phi_i^2}-\sum_{j=1}^{k_s}\frac{(s_j-s_j')^2}{c_j^2}-\frac{(t-t')^2}{a^2}\right), \qquad (4.3)$$

where ϕ_i, c_i and a are dependence range parameters of input, space and time respectively. (4.3) assumes the separability in input, space and time, because it implies $\rho(\mathbf{x}_i,\mathbf{x}_j) = \rho_\xi(\boldsymbol{\xi}_i,\boldsymbol{\xi}_j)\rho_s(\mathbf{s}_i,\mathbf{s}_j)\rho_t(t_i,t_j)$. When computer model outputs are generated on a $n_\xi\times n_s\times n_t$ regular grid of input, space and time, and data are ordered properly, we have $R = R_\xi\otimes R_s\otimes R_t$, where $R_\xi = [\rho_\xi(\boldsymbol{\xi}_i,\boldsymbol{\xi}_j)]_{i,j=1,\dots,n_\xi}, R_s = [\rho_s(s_i,s_j)]_{i,j=1,\dots,n_s}, R_t = [\rho_t(t_i,t_j)]_{i,j=1,\dots,n_t}$. Since $R^{-1} = R_\xi^{-1}\otimes R_s^{-1}\otimes R_t^{-1}$ and $|R| = |R_\xi|^{n_s n_t}|R_s|^{n_\xi n_t}|R_t|^{n_\xi n_s}$, the computations of evaluating the likelihood can be greatly reduced when n is very large.

Although the computer model is deterministic, a small variance term $\tau^2\delta_{\mathbf{x}=\mathbf{x}'}$ is usually added to $\rho(\mathbf{x},\mathbf{x}';\boldsymbol{\theta})$ for numerical stability, where δ is the Kronecker delta function. This small variance term is also referred to as the "nugget" effect in spatial statistics, accounting for the measurement error. We incorporate the nugget effect parameter τ^2 in $\boldsymbol{\theta}$ throughout the paper for simplicity. For auto-correlation model that is separable in input, space and time, three nugget terms $\tau_\xi^2 \delta_{\xi_i = \xi_j}$, $\tau_s^2 \delta_{\mathbf{s}_i = \mathbf{s}_j}$, and $\tau_t^2 \delta_{t_i = t_j}$ are added to $\rho_\xi(\cdot, \cdot)$, $\rho_s(\cdot, \cdot)$, and $\rho_t(\cdot, \cdot)$ respectively. Adding nuggets in this way can preserve the separability so that fast computations can still be achieved.

4.2.2 Nonseparable auto-correlation models

Although the separable auto-correlation model is easy to construct and leads to reduced computational costs, it may be too restrictive in some applications due to its implications on the covariance dependence structures. Take a computer model output $f(\cdot, \cdot)$ with two dimensional input parameters (ξ, t) as an example, the separable covariance implies that

$$\mathrm{Corr}((f(\xi, t), f(\xi', t'))|f(\xi, t')) = 0,$$

under the Gaussian process assumption [45, 55]. This implication on conditional correlation may be too restrictive in some applications. Besides, the separable structure also implies that

$$\mathrm{Corr}(f(\xi, t), f(\xi, t')) = \rho_t(t, t'),$$

which means the correlation structure of dimension t can not vary over the dimension ξ. [55] showed that if a process $f(\xi, t)$ has a separable covariance function, then $f(\xi, t)$ is second-order identical to the product of second-order uncorrelated processes $f^{(1)}(\xi)$ and $f^{(2)}(t)$. More comprehensive discussions of the implications of separable covariance function for emulation can be found in[55].

Therefore, the nonseparable auto-covariance model may be preferred when these assumptions are not true for the dataset. We propose to use the nonseparable covariance functions [12, 30, 66] for the Gaussian process emulator. These models are more general and include separable covariance functions as special cases. One example of nonseparable correlation functions in input and time is

$$\rho(\mathbf{x}, \mathbf{x}') = \left(\frac{|v|^{2\alpha}}{a} + 1 \right)^{-\frac{k_\xi}{2}} \exp\left(- \frac{\sqrt{\sum_{i=1}^{k_\xi} |u_i|^2/\phi_i^2}}{\left(\frac{|v|^{2\alpha}}{a} + 1 \right)^{\beta/2}} \right), \tag{4.4}$$

where $u_i = |\xi_i - \xi_i'|$, $v = |t - t'|$; $a > 0$ is the dependence range parameter in time, $\phi_i > 0$ is the dependence range parameter for i^{th} input dimension, $\alpha \in (0, 1]$ is the smoothness parameter and $\beta \in [0, 1]$ is the input-time interaction parameter. When $\beta = 0$, it reduces to the separable case. In this paper, we construct a more sophisticated nonseparable model in input, space and time following the work in [1],

$$\begin{aligned}
\rho(\mathbf{h}, \mathbf{u}, v) = {} & \left(a_1 \left(\frac{\|\mathbf{u}\|^2}{(a_4|v|^{\alpha_4} + 1)^{\beta_4}} \right)^{\alpha_1} + 1 \right)^{-\beta_1 k_s/2} (a_2|v|^{2\alpha_2} + 1)^{-\beta_2 k_\xi/2} \\
& \times (a_3\|\mathbf{h}\|^{2\alpha_3} + 1)^{-\beta_3/2} (a_4|v|^{2\alpha_4} + 1)^{-\beta_4 k_\xi/2} \exp\left(- \frac{c_1\|\mathbf{h}\|^{2\gamma_1}}{\left(a_1 \left(\frac{\|\mathbf{u}\|^2}{(a_4|v|^{2\alpha_4}+1)^{\beta_4}} \right)^{\alpha_1} + 1 \right)^{\beta_1\gamma_1}} \right) \\
& \times \exp\left(- \frac{c_2\|\mathbf{u}\|^{2\gamma_2}}{(a_2|v|^{2\alpha_2} + 1)^{\beta_2\gamma_2}} - \frac{c_3|v|^{2\gamma_3}}{(a_3\|\mathbf{h}\|^{2\alpha_3} + 1)^{\beta_3\gamma_3}} \right),
\end{aligned} \tag{4.5}$$

where $\mathbf{h} = \|\mathbf{s} - \mathbf{s}'\|, \mathbf{u} = \|\boldsymbol{\xi} - \boldsymbol{\xi}'\|$ and $v = |t - t'|$. To ensure the positive-definiteness of (4.5), we need $c_i > 0, \gamma_i \in (0, 1]$ for $i = 1, 2, 3$, and $a_j > 0, \alpha_j \in (0, 1], \beta_j \in [0, 1]$ for $j = 1, \ldots, 4$. The parameters $\alpha_1, \ldots, \alpha_4$ and $\gamma_1, \gamma_2, \gamma_3$ can be interpreted as smoothness parameters; $a_1, \ldots, a_4$ and c_1, c_2, c_3 are scale parameters; $\beta_1, \ldots, \beta_4$ are interaction parameters, modeling the two-way and three-way interactions among input, space and time. If we fix the smoothness parameters $\gamma_j = 0.5, j = 1, 2, 3$, then it is more clearly to see that β_j's determine the interaction effects. Although the model in (4.5) is very flexible, it involves too many unknown model parameters. In this paper, we will focus on the nonseparable auto-correlation model between two components of input, space and time, with similar form to (4.4).

4.2.3 FSA-Block approximation

Since the computations of R^{-1} and $|R|$ in (4.2) become expensive or even infeasible when n is large, we need to employ some computational techniques to overcome the computation bottleneck. There are several existing methods to facilitate computations of the Gaussian process model that rely on covariance approximations. Popular covariance approximation models include the Gaussian predictive process [4, 21], the fixed rank kriging model [13], and the covariance tapering [43, 23], to name a few.

In this work we propose to use the Full-Scale Approximation with Block modulating function (FSA-Block) to speed up computations [59, 60]. It consists of a summation of a reduced rank covariance and a sparse covariance with the block diagonal structure. This approach combines the merits of both reduced rank and sparse covariances without adding much computational complexity.

In the following we describe the FSA-Block approach for a Gaussian process with zero-mean, unit variance and a correlation function $\rho(\cdot, \cdot)$. The FSA-Block approximation is motivated by the Karhunen-Loéve orthogonal expansion (K-L expansion) of the Gaussian Process, which decomposes a covariance function as:

$$\rho(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{\infty} \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}'), \tag{4.6}$$

where λ_i are the eigenvalues of the process, $\psi_i(\mathbf{x})$ are the corresponding orthonormal eigenfunctions; the eigenvalue-eigenfunction pairs are solutions to the integral equation $\int_D \rho(\mathbf{x}, \mathbf{t})\psi_i(\mathbf{t})d\mathbf{t} = \lambda_i\psi_i(\mathbf{x})$.

The leading terms in (4.6) are often assumed to capture the main feature of the covariance and thus the residual terms are typically dropped from the expansion to yield a reduced rank approximation of the covariance. Although increasing the rank can preserve more information about the fine scale covariance pattern, computations become more expensive. Motivated from the decomposition in (4.6), we give a more careful treatment of the covariance that can preserve most information present in both the leading reduced rank terms and the residual covariance yet still achieves computational efficiency.

Solving the integral equation for the K-L expansion is typically a challenging task. We use the Nyström discretization [2], a numerical method for solving integral equations, to approximate the reduced rank part of the K-L decomposition. Consider a set of knots $\mathscr{X}^* = \{\mathbf{x}_1^*, \ldots, \mathbf{x}_{n^*}^*\}$. Let R_{**} denote the $n^* \times n^*$ correlation matrix whose (i, j) entry is $\rho(\mathbf{x}_i^*, \mathbf{x}_j^*)$. Let $\{\mathbf{u}_i^{(n^*)}\}$ and $\{\lambda_i^{(n^*)}\}$ be the eigenvectors and the eigenvalues for the correlation matrix R_{**}. The Nyström approximation of the leading n^* eigenfunctions and eigenvalues for the correlation kernel $\rho(\mathbf{x}, \mathbf{x}')$ are

$$\psi_i(\mathbf{x}) \approx \frac{\sqrt{n^*}}{\lambda_i^{(n^*)}}\rho(\mathbf{x}, \mathscr{X}^*)\mathbf{u}_i^{(n^*)}, \quad \lambda_i \approx \frac{\lambda_i^{(n^*)}}{n^*}, \quad \text{for} \quad i = 1, \cdots, n^*.$$

It can be proved that the Nyström approximation method leads to a reduced rank correlation

$$\rho_{pp}(\mathbf{x}, \mathbf{x}') = \rho(\mathbf{x}, \mathscr{X}^*)R_{**}^{-1}\rho^T(\mathbf{x}', \mathscr{X}^*),$$

where $\rho(\mathbf{x}, \mathscr{X}^*) = [\rho(\mathbf{x}, \mathbf{x}')]_{\mathbf{x}' \in \mathscr{X}^*}$. Using the reduced rank model derived from the Nyström approximated correlation, $\rho(\mathbf{x}, \mathbf{x}') - \rho_{pp}(\mathbf{x}, \mathbf{x}')$ remains to be positive semi-definite following the Schur complement property in linear algebra. We approximate it by multiplying the residual correlation with a modulating function $\mathcal{K}(\mathbf{x},\mathbf{x}')$, that is, $\rho_\epsilon(\cdot,\cdot) = (\rho(\cdot,\cdot) - \rho_{pp}(\cdot,\cdot))\mathcal{K}(\cdot,\cdot)$. The modulating function has to be chosen to ensure $\rho_\epsilon(\mathbf{x},\mathbf{x}')$ is positive semi-definite. We also assume it has the property of having zero entries for a large proportion of possible location pairs $(\mathbf{x},\mathbf{x}')$ so that $\rho_\epsilon(\cdot,\cdot)$ evaluated on $\mathscr{X}$ is a sparse matrix. One specific choice of $\mathcal{K}(\cdot,\cdot)$ is the block modulating function. Given a partition of observed locations $\cup_{i=1}^K B_i = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\} = \mathscr{X}$, it is defined as

$$\mathcal{K}_{block}(\mathbf{x},\mathbf{x}') = \begin{cases} 1 & \text{if } \mathbf{x},\mathbf{x}' \in B_i, i = 1,\ldots,K; \\ 0 & \text{Otherwise.} \end{cases}$$

If observations are grouped together within each B_i, the $\rho_\epsilon(\cdot,\cdot)$ on $\mathscr{X}$ yields a block-diagonal matrix whose inverse can be computed easily.

The FSA-Block method approximates the parent correlation function ρ by the sum of $\rho_{pp}(\cdot,\cdot)$ and an approximated residual correlation function $\rho_\epsilon(\cdot,\cdot)$:

$$\rho^\dagger(\mathbf{x},\mathbf{x}') = \rho_{pp}(\mathbf{x},\mathbf{x}') + \rho_\epsilon(\mathbf{x},\mathbf{x}'),$$

which is still a valid correlation function. Under this correlation approximation, R is approximated by $R^\dagger = R_{nn^*}R_{**}^{-1}R_{nn^*}^T + R_\epsilon$, where $R_{nn^*} = \rho(\mathscr{X},\mathscr{X}^*)$ and $R_\epsilon = \rho_\epsilon(\mathscr{X},\mathscr{X})$. The Sherman-Woodbury-Morrison inversion formula yields

$$R^{\dagger-1} \;=\; R_\epsilon^{-1} - R_\epsilon^{-1}R_{nn^*}(R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**})^{-1} R_{nn^*}^T R_\epsilon^{-1}. \qquad (4.7)$$

Thus $R^{\dagger-1}$ involves the calculations of inverses of a block diagonal matrix R_ϵ and a $n_* \times n_*$ matrix $R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**}$. If we choose n^* and block size to be small, the computations of R^{-1} using $R^{\dagger -1}$ can be greatly reduced. Indeed, the computational complexity of calculating $R^{\dagger -1}$ is $\mathcal{O}(nn^{*2} + nn_b^2)$, where n_b is the average block size. The determinant of $R^\dagger$ can also be computed efficiently using Sylvester's determinant theorem

$$|R^\dagger| = |R_\epsilon||R_{**}|^{-1}|R_{**} + R_{nn^*}^T R_\epsilon^{-1} R_{nn^*}|. \tag{4.8}$$

So instead of computing the determinant of a big $n \times n$ matrix, we only need to compute the determinants of a $n^* \times n^*$ matrix and a block-diagonal matrix.

As described above, fast computations can be achieved using the FSA-Block approach. The correlation function of the FSA-Block approach is

$$\rho^\dagger(\mathbf{x}, \mathbf{x}') = \begin{cases} \rho(\mathbf{x}, \mathbf{x}') & \text{if } \mathbf{x}, \mathbf{x}' \in B_i, i = 1, \ldots, K; \\ \rho_{pp}(\mathbf{x}, \mathbf{x}') & \text{otherwise.} \end{cases}$$

Therefore, the correlation within blocks are preserved exactly and the correlation across blocks are approximated by that of the reduced rank part. Since the FSA-Block approach provides a general way of approximating any given covariance functions without further restrictions on the parent covariance structures, it also applies to the separable auto-covariance model if computations in certain dimensions are infeasible due to large sample sizes in those dimensions. For example, if learning the response on a highly fine-resolution spatial grid is desirable in certain studies, the FSA-Block approach can be applied only to a spatial correlation function to facilitate computations.

4.3 Bayesian Inference of Model Parameters and Prediction

4.3.1 Prior specifications

For the multivariate Gaussian process regression model, the unknown parameter set is $\{B, \Sigma, \boldsymbol{\theta}\}$. We assume the prior distributions of $\{B, \Sigma\}$ and $\boldsymbol{\theta}$ are independent, namely $\pi(B, \Sigma, \boldsymbol{\theta}) = \pi(B, \Sigma)\pi(\boldsymbol{\theta})$. For $\pi(B, \Sigma)$, we assign a noninformative conjugate prior

$$\pi(B, \Sigma) \propto |\Sigma|^{-\frac{q+1}{2}}. \tag{4.9}$$

The prior specification of $\pi(\boldsymbol{\theta})$ depends on the specific form of the covariance function. Customarily, the inverse-gamma prior can be assigned to the nugget τ^2; the input (spatial/temporal) range parameter can be assigned with a reasonably informative prior, e.g. an uniform prior with its support specified according to the belief of the practical input (spatial/temporal) dependence range of the computer model outputs; for the smoothness parameter, a uniform prior with a reasonable support reflecting prior information about the smoothness of the process can be assigned.

4.3.2 Bayesian inference on the model parameters

In the Bayesian framework, inference on the parameters of the model is performed through the posterior distribution which can be computed according to the Bayes theorem. Here, the density of the posterior distribution is not available in closed form. However, we can resort to the MCMC methods in order to perform inference. We consider an MCMC sampler which consists of updating blocks of $(B, \Sigma|Y, \theta)$ and $(\theta|Y)$ separately. The pseudo-code of one sweep of this MCMC sampler in given in Algorithm 1. A sketch of the derivation of the conditional posterior distributions involved in the MCMC sampler is given below.

Algorithm 1 Blocks of the MCMC sampler

```
• Update $\boldsymbol{\theta}$ by using a Metropolis-Hastings within Gibbs algorithm [68, 36] targeting $\pi(\theta|Y)$.
  For $i = 1, ..., d_\theta$:
    1. Draw $\boldsymbol{\theta}_i'$ from a proposal distribution $Q_i(\cdot|\boldsymbol{\theta})$, where $Q_i(\cdot|\cdot)$ is pre-defined by the researcher, and set $\boldsymbol{\theta}' = (\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_{i-1}, \boldsymbol{\theta}_i', \boldsymbol{\theta}_{i+1}, ..., \boldsymbol{\theta}_{d_\theta})$.
    2. Accept $\boldsymbol{\theta}'$ as the next state with prob. $\min\{1, \frac{\pi(\boldsymbol{\theta}'|Y)}{\pi(\boldsymbol{\theta}|Y)} \frac{Q_i(\boldsymbol{\theta}_i|\boldsymbol{\theta}')}{Q_i(\boldsymbol{\theta}_i'|\boldsymbol{\theta})}\}$.
• Update $(B, \Sigma)$, by sampling directly as
    – Sample $\Sigma|Y, \boldsymbol{\theta}$ from $\mathcal{IW}(n - m, Y^T R^{-1} Y - \hat{B}_{gls}^T (H^T R^{-1} H) \hat{B}_{gls})$.
    – Sample $B|Y, \Sigma, \boldsymbol{\theta}$ from $\mathcal{N}_{m \times q}(\hat{B}_{gls}, (H^T R^{-1} H)^{-1}, \Sigma)$.
```

According to the Bayes theorem, the density of the joint posterior distribution of $(B, \Sigma, \boldsymbol{\theta}|Y)$ is such that

$$\begin{aligned} \pi(B, \Sigma, \boldsymbol{\theta}|Y) \propto |R|^{-q/2} |\Sigma|^{-n/2} \exp\left(-\frac{1}{2} \mathrm{tr}\left(\Sigma^{-1}(Y - HB)^T R^{-1}(Y - HB)\right)\right) \\ \times |\Sigma|^{-\frac{q+1}{2}} \times \pi(\boldsymbol{\theta}). \end{aligned} \tag{4.10}$$

The joint posterior density (4.10) is not available in closed form. However, it is straightforward to show that $\pi(B, \Sigma, \boldsymbol{\theta}|Y) = \pi(B, \Sigma|Y, \boldsymbol{\theta}) \times \pi(\boldsymbol{\theta}|Y)$. Therefore, we opt to sample from $\pi(\boldsymbol{\theta}|Y)$ and $\pi(B, \Sigma|Y, \boldsymbol{\theta})$ separately.

The conditional posterior distribution of $(B, \Sigma|Y, \boldsymbol{\theta})$ is of standard form $\pi(B, \Sigma|Y, \theta) = \pi(B|Y, \Sigma, \boldsymbol{\theta}) \times \pi(\Sigma|Y, \boldsymbol{\theta})$, such that

$$B|Y, \Sigma, \boldsymbol{\theta} \sim \mathcal{N}_{m \times q}(\hat{B}_{gls}, (H^T R^{-1} H)^{-1}, \Sigma); \tag{4.11}$$

$$\Sigma|Y, \boldsymbol{\theta} \sim \mathcal{IW}(n - m, Y^T R^{-1} Y - \hat{B}_{gls}^T (H^T R^{-1} H) \hat{B}_{gls}), \tag{4.12}$$

where $\mathcal{N}_{m\times q}$ stands for the matrix-normal distribution, $\mathcal{IW}$ stands for the Inverse Wishart distribution, and $\hat{B}_{gls}$ is the generalized least squares estimator of B, i.e., $\hat{B}_{gls} = (H^T R^{-1} H)^{-1} H^T R^{-1} Y$. This convenient form results allow us to sample directly from the conditional posterior $\pi(B, \Sigma | Y, \theta)$.

By integrating out the hyper-parameter matrices B and Σ in (4.10), it is easy to show that the marginal posterior distribution of $\boldsymbol{\theta}|Y$ is

$$\pi(\boldsymbol{\theta}|Y) \propto \pi(\theta)|R|^{-\frac{q}{2}}|H^T R^{-1} H|^{-\frac{q}{2}}|(Y - H\hat{B}_{gls})^T R^{-1}(Y - H\hat{B}_{gls})|^{-\frac{n-m}{2}}. \quad (4.13)$$

The marginal posterior $\pi(\boldsymbol{\theta}|Y)$ cannot be sampled directly, however its density (4.13) is known up to a normalizing constant. Therefore, one can draw samples from $\pi(\boldsymbol{\theta}|Y)$ by using a Metropolis-Hastings within Gibbs algorithm [68, 36] sampler. A log-normal distribution or a truncated normal distribution centered at the current value can be used as the proposal distribution $Q_i(\cdot; \boldsymbol{\theta})$ in Algorithm 1.

It is worth mentioning that, when the sample size n is large, we can replace R in (4.11), (4.12), and (4.13) with the FSA-Block approximated correlation matrix $R^\dagger$ introduced in Section 2.3, and then apply the inversion formula (4.7) to efficiently calculate the inverse of $R^\dagger$. The details of calculating the approximated posterior distributions by the FSA-Block approach are given in the Appendix.

Regarding Algorithm 1, in practice, instead of running the whole sweep recursively, one can run the first block recursively in order to collect a sample from $\pi(\boldsymbol{\theta}|Y)$ at first stage, and then use these draws in order to sample from $\pi(B, \Sigma|Y, \theta)$ at second stage. In cases where parallel computing environment is available, this may reduce the computational cost of sampling.

4.3.3 Prediction

The Bayesian predictive distribution provides a natural measure on the function space of surrogate models. At a new point $\mathbf{x}_p$, the predictive distribution has its mean

$$E(\mathbf{f}(\mathbf{x}_p)|B, \boldsymbol{\theta}, Y) = \mathbf{h}^T(\mathbf{x}_p)B + R_{p,n}R^{-1}(Y - HB) \tag{4.14}$$

and variance

$$\mathrm{Var}(\mathbf{f}(\mathbf{x}_p)|B, \Sigma, \boldsymbol{\theta}, Y) = (1 - R_{p,n}R^{-1}R_{p,n}^T) \otimes \Sigma, \tag{4.15}$$

where $R_{p,n} = [\rho(\mathbf{x}_p, \mathbf{x})]_{\mathbf{x}\in\mathscr{X}}$ is the $1 \times n$ correlation vector. When n is too large so that R^{-1} is computationally prohibitive, we can again apply the FSA-Block approximation method to approximating R with $R^{\dagger}$ whose inversion can be done efficiently using (4.7).

In uncertainty quantification, interest lies especially on the means and the associated error bars of response surfaces. We can obtain a sample of the mean response of the i^{th} output by integrating out the input parameters $\boldsymbol{\xi}$ in $\mathbf{x}_p$ in (4.14),

$$M_p^i = \bar{\mathbf{h}}_p^T B_i + \bar{R}_p R^{-1}(Y_i - HB_i), \tag{4.16}$$

where $Y_i = (f_i(\mathbf{x}_1), \ldots, f_i(\mathbf{x}_n))^T$, $B_i = (B_{1i}, \ldots, B_{mi})^T$, $\bar{h}_p = \int \mathbf{h}(\mathbf{x}_p)p(\boldsymbol{\xi})d\boldsymbol{\xi}$, $\bar{R}_p = \int R_{p,n}p(\boldsymbol{\xi})d\boldsymbol{\xi}$, and $p(\boldsymbol{\xi})$ is the joint density of input variables. The covariance of M_p^i is

$$V_p^i = \int E(f_i(\mathbf{x}_p)|B_i, \boldsymbol{\theta}, Y_i)E(f_i(\mathbf{x}_p)|B_i, \boldsymbol{\theta}, Y_i)^T p(\boldsymbol{\xi})d\boldsymbol{\xi} - M_p^i M_p^{iT}. \tag{4.17}$$

In the case when (4.16) and (4.17) are not available in closed forms, the integrals can be approximated by the Monte Carlo method on a dense grid of the input space.

4.4 Numerical Results

4.4.1 2-input and 1-output example

We use a simulation example to show that the nonseparable covariance model can outperform the separable model in some cases. We consider the following function to generate output data,

$$f(x_1, x_2) = x_1 \exp\left(-\sqrt{x_1^2 + x_2^2}\right),$$

where the inputs $x_1, x_2 \in [-6, 6]$. In this case $f(x_1, x_2) \neq f_1(x_1)f_2(x_2)$, and hence the separable covariance model may not be adequate according to the theory in [55].

We then use a Gaussian process model in (4.1) with constant means as a surrogate to fit the simulated function values. We consider the following nonseparable covariance model [30],

$$\mathcal{C}(f(x_1, x_2), f(x_1', x_2')) = \frac{\sigma^2}{\left(\frac{v^{2\alpha}}{a} + 1\right)} \exp\left(-\frac{u^2}{c\left(\frac{v^{2\alpha}}{a} + 1\right)^{\beta}}\right), \tag{4.18}$$

where $u = |x_1 - x_1'|$ and $v = |x_2 - x_2'|$. We also consider two separable models for comparisons; (a). the covariance model as in (4.18) but with the interaction parameter $\beta = 0$ (denoted by "Sep"), and (b). the commonly used squared exponential covariance model (denoted by "Sqexp"). A fixed small nugget effect $\tau^2 = 10^{-6}$ was added to the covariance function for numerical stability. We experimented with different sample sizes n for the training set. We also fixed a prediction set and evaluate the mean squared prediction errors (MSPE) to compare the prediction performance of different covariance models. Specifically, the training sets were $n = 200, 500$ function values evaluated at locations selected by Latin Hypercube Sampling (LHS). The prediction set was fixed to be 100 function values evaluated at hold-out locations selected by LHS. Uniform priors with a reasonable support were assigned to the dependence range parameters a and c; the uniform prior on $[0, 1]$ was assigned to the smoothness parameter α and interaction parameter β in (4.18). We collected 6000 posterior samples after a burn-in period of 1000 iterations. Then the posterior means of model parameters were plugged in (4.14) to obtain prediction results. The parameter estimation and prediction results are summarized in Table 4.1. First we observed that in both experimental cases, the posterior mean estimates of β of the nonseparable model are far from zero, indicating the existence of the interaction effect. Also for both cases, the nonseparable model outperforms the separable models in terms of the prediction. For relatively large sample size $n = 500$, the estimation of β becomes more accurate (posterior variance reduces from 0.021 to 0.008) and the prediction results of the nonseparable model are obviously better than those of the two separable covariance models.

Table 4.1: Posterior means and MSPEs.

		a	c	α	β	B_0	σ^2	MSPE
$n = 200$	Nonsep	3.728	4.633	1.000	0.641	-0.001	0.0061	$1.67 \cdot 10^{-5}$
	Sep	2.978	3.711	1.000	0	$-6.59 \cdot 10^{-5}$	0.0053	$3.67 \cdot 10^{-5}$
	Sqexp	1.105	1.984	–	–	$-4.28 \cdot 10^{-4}$	0.0033	$3.88 \cdot 10^{-5}$
$n = 500$	Nonsep	2.693	2.314	0.990	0.721	$2.30 \cdot 10^{-4}$	0.0039	$4.28 \cdot 10^{-7}$
	Sep	2.246	1.987	0.979	0	$3.28 \cdot 10^{-5}$	0.0033	$1.57 \cdot 10^{-6}$
	Sqexp	0.933	1.016	–	–	$3.31 \cdot 10^{-5}$	0.0023	$9.58 \cdot 10^{-6}$

4.4.2 Krainchnan-Orszag three-mode problem

In this example, we consider the system of ordinary differential equations with respect to t as in [74],

$$\begin{aligned}\frac{dy_1}{dt} &= y_1 y_3, \\ \frac{dy_2}{dt} &= -y_2 y_3, \\ \frac{dy_3}{dt} &= -y_1^2 + y_2^2,\end{aligned}$$

subject to stochastic initial conditions $y_1(0) = 1, y_2(0) = 0.1\xi_1, y_3(0) = \xi_2$, where $\xi_i \sim U(-1, 1), i = 1, 2$. This problem has 2 input variables and 3 outputs. It is of interest because the response has a discontinuity line at $\xi_1 = 0$, inducing a nonstationary response surface in input space. Here we applied the stationary nonseparable covariance model to this problem with a relatively large sample size n to obtain reasonable prediction results. However the Bayesian inference is computationally intensive due to large sample size, hence the FSA-Block approach was applied to the nonseparable model for computational efficiency.

The training set was obtained at 600 input points selected by Latin Hypercube Sampling on a time grid $T = 1, 2, \ldots, 10$. We considered the validation set of a 31×31 input grid on time points 11 and 12, which allows us to assess prediction performance in both input and time scenario. The multivariate Gaussian process model in (4.1) with constant means were used to fit for each output $y_i(t), i = 1, 2, 3$. The nonseparable auto-correlation function considered was the model in (4.4) with $k_\xi = 2$, and the same two separable models as in the previous simulation study were used for comparison purpose. A fixed nugget $\tau^2 = 10^{-6}$ was added to the covariance model to improve numerical stability. When implementing the nonseparable model, we applied the FSA-Block approach with 20 input knots selected by LHS at each time grid point (in total 200 knots) and 30 blocks created by K-means clustering algorithm. The prior specifications of model parameters were similar to those in the previous example. After a burn-in period of 1000 iterations, we collected 6000 posterior samples for inference. The posterior means of model parameters were plugged in (4.14) to obtain the predictive response surface.

Table 4.2: Posterior means of model parameters and MSPEs.

$n_\xi = 600$	Nonsep	Sep	Sqexp
a	3.306	1.315	1.265
c_1	5.748	6.540	4.608
c_2	21.648	22.796	16.351
α	0.999	0.999	–
β	0.996	0	–
B_{10}	0.478	0.461	0.474
B_{20}	−0.106	0.002	−0.006
B_{30}	−0.044	−0.209	−0.197
Σ_{11}	0.296	0.505	0.312
Σ_{12}	0.007	−0.001	0.0003
Σ_{13}	−0.017	−0.047	−0.033
Σ_{22}	1.257	1.187	0.768
Σ_{23}	0.108	0.114	0.072
Σ_{33}	1.567	1.729	1.107
$\text{MSPE}_{sp\&t}$	0.307	0.337	0.325

Table 4.2 shows the parameter estimations and prediction results of the three covariance models. The posterior mean of β by the nonseparable model is very close to 1, suggesting that modeling the interaction between input and time may be beneficial. The separable model in (4.4) with $\beta = 0$ produced parameter estimates close to those from the nonseparable model, except for a much smaller estimate of the time dependence parameter. In terms of the prediction, the nonseparable model obviously outperforms the two separable models. Specifically, it produces MSPEs

$(0.078, 0.233)$ for $y_1(t)$ and $y_2(t)$ respectively, which are significantly smaller than $(0.090, 0.247)$ by the separable model in (4.4) with $\beta = 0$ and $(0.092, 0.290)$ by the squared exponential model; for $y_3(t)$, the nonseparable model has a comparable result to that of separable models.

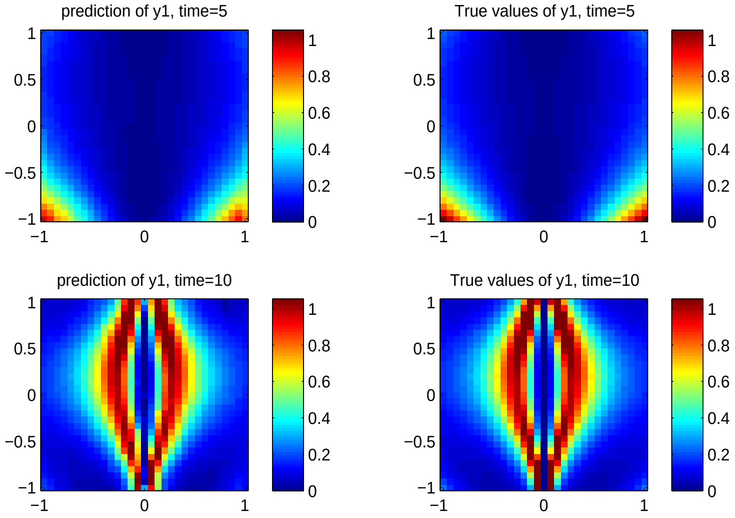


Figure 4.1: Predictive surface of input space of y_1 at time points 5 and 10 using the nonseparable model.

We then checked the predictive input surface of each output by the nonseparable model at selected time points, and reported the results in Figure 4.1, Figure 4.2 and Figure 4.3, respectively. We randomly chose the results at time points 5 and 10 for illustrations. For $y_1(t)$ and $y_2(t)$, the predictive input surfaces by the nonseparable model are very close to the true response surfaces in general, but the prediction errors are high around $\xi_1 = 0$ for $y_3(t)$ at time point 10. Figure 4.4 shows the MSPE surfaces of 3 outputs in input space by averaging MSPEs over time, and it is more clear that the prediction errors of $y_3(t)$ peaked at $\xi_1 = 0$. Recall that here the computer model

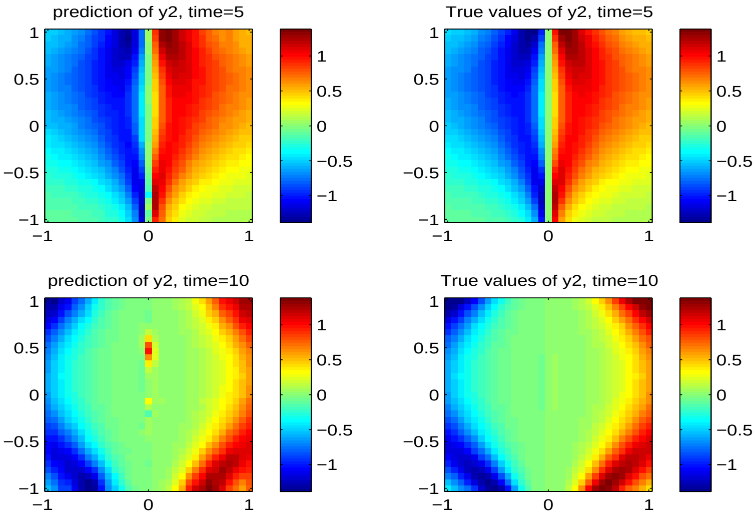


Figure 4.2: Predictive surface of input space of y_2 at time points 5 and 10 using the nonseparable model.

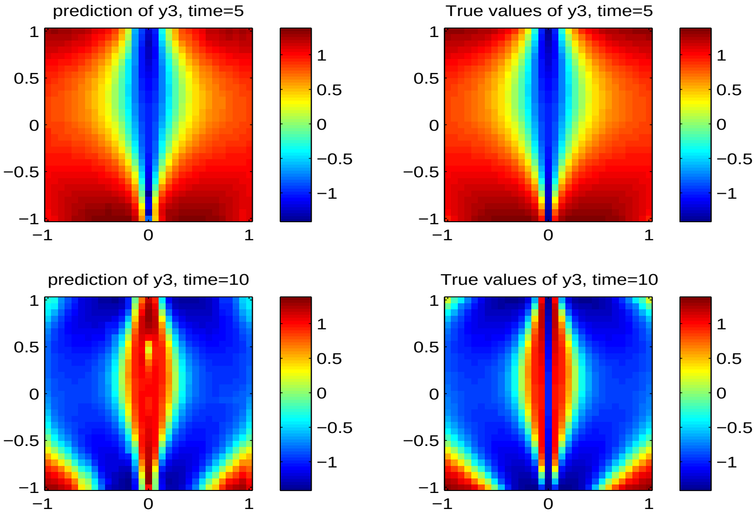


Figure 4.3: Predictive surface of input space of y_3 at time points 5 and 10 using the nonseparable model.

outputs have a discontinuous point at $\xi_1 = 0$. Therefore, the stationary covariance function may not be adequate to model this nonstationary feature. In order to have better prediction results, we need more observations sampled around $\xi_1 = 0$ for the stationary covariance models or consider a nonstationary covariance model. We also checked the predictive input surfaces of 3 outputs by the 2 separable models, and the results are similar to the nonseparable model results.

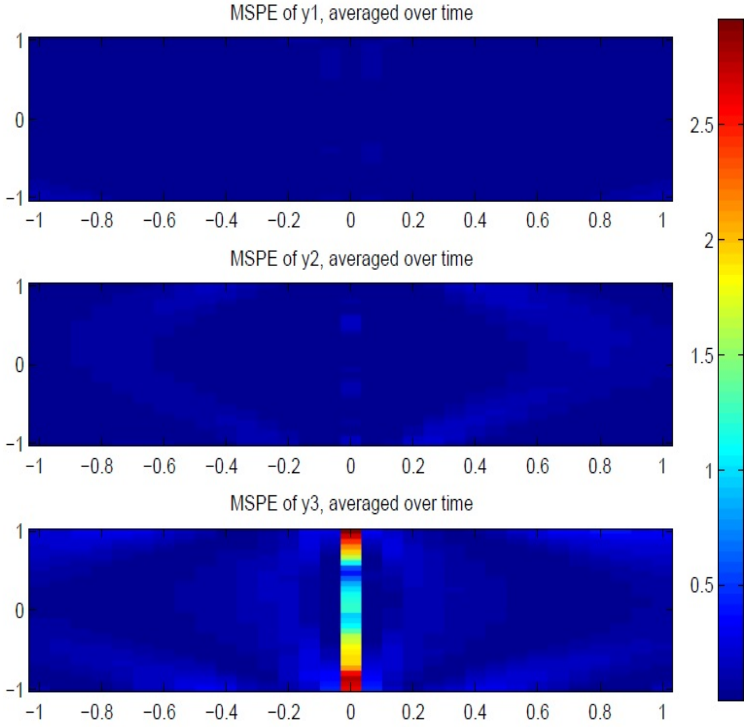


Figure 4.4: The MSPEs averaged over time by the nonseparable model.

The predictive mean time curve is also of crucial interest, which shows the shape of the mean response curve averaged over input space. We obtained the predictive mean response curve of t for each output over 100 LHS selected input points. Then predictions were done at 50 equidistant time steps in $[0, 10]$ and it was repeated for

100 times to obtain the error bars of the predictive mean time curve. To calculate the mean of a sampled response curve, we used the method in (4.16). The results of computer code outputs at integer time grid were used as baselines. Figure 4.5 shows the predictive mean curves for $y_1(t)$ and $y_3(t)$ as a function of time, as well as their corresponding 95% confidence intervals using the nonseparable model. We can see that the error bars of the predictive mean curves are tight and can cover the true means of the computer code outputs, indicating the effectiveness of the nonseparable model.

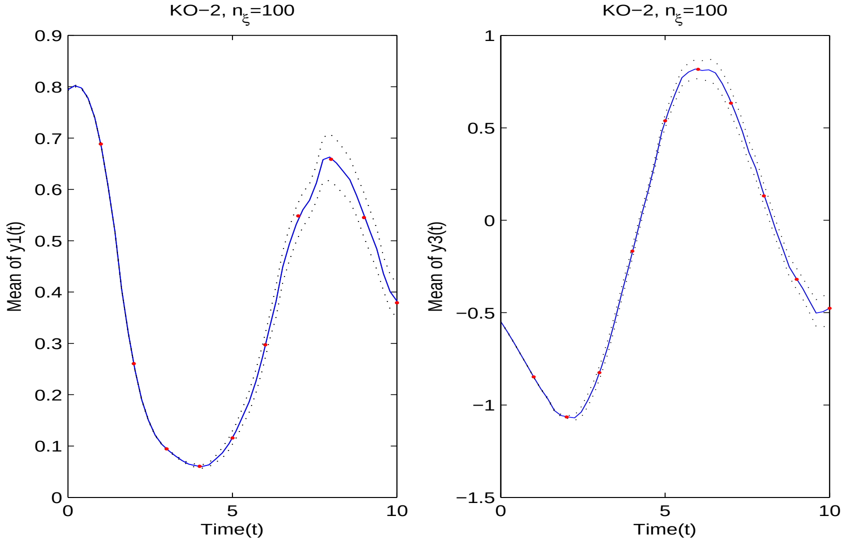


Figure 4.5: The predictive mean curve in time by the nonseparable model. The blue line is the predictive mean curve of time; the dash lines are corresponding 95% confidence intervals; the red dots are the means of the computer code outputs.

4.4.3 Flow through porous media example

We use this example to show the effectiveness of our method in modeling large computer code outputs. The proposed Gaussian process surrogate model was applied to a petroleum reservoir simulation of a much larger data size. The object is a two-dimensional, single phase, steady flow through a random permeability field. The spatial domain $\mathscr{X}_s = [0,1]^2$, representing an idealized oil reservoir. The pressure p and the velocity fields of the flow $\mathbf{u}$ are of key interests and they are connected via Darcy law: $\mathbf{u} = -\mathbf{K}\nabla p$ in $\mathscr{X}_s$, where $\mathbf{K}$ is the permeability tensor. The pressure p satisfies $-\nabla \cdot (\mathbf{K}\nabla p) = f$ in $\mathscr{X}_s$, where f may be used to model the injection/production wells. One specific choice of f is

$$f(\mathbf{s}) = \begin{cases} -r, & \text{if} |s_i - w/2| < w/2, \text{ for } i = 1,2, \\ r, & \text{if} |s_i - 1 + w/2| < w/2, \text{ for } i = 1,2, \\ 0, & \text{otherwise.} \end{cases}$$

There are two wells at the lower-left and upper-right corners of the spatial domain, with r and w specifying well rates and well sizes respectively ($r = 10$ and w=1/8 in this case). After specifying the permeability tensor $\mathbf{K}$ and imposing certain boundary conditions, the pressure p and the velocities $\mathbf{u}$ can be solved numerically.

Specifically, $\mathbf{K}$ is assumed to be isotropic $K_{ij} = K\delta_{ij}$ and modeled by a log-Gaussian process: $K(\mathbf{s}) = \exp(G(\mathbf{s}))$, where δ is the Kronecker delta function and $G(\cdot) \sim \mathcal{GP}(m_G, \mathcal{C}_G(\cdot,\cdot))$. The covariance function $\mathcal{C}_G(\cdot,\cdot)$ is the separable exponential covariance. The truncated Karhunen-Loève expansion on $G(\cdot)$ gives its finite dimension representation

$$K(\boldsymbol{\xi}; \mathbf{s}) = \exp\left(m_G + \sum_{k=1}^{p} w_k \psi_k(\mathbf{s}) \right),$$

where w_k are uncorrelated standard Gaussian random variables and $\psi_k(\mathbf{s})$ are eigenfunctions of the exponential covariance function $\mathcal{C}_G(\cdot,\cdot)$. Then uniform variables $\xi_k = \Phi(w_k) \sim U([0,1])$ are treated as input variables. More details of this example can be found in [7]. In this example, we truncate $G(\cdot)$ after $p = 50$ terms.

The training set is on a $24 \times 32 \times 32$ input-spatial grid, so the total training data has 24576×3 observations. The nonseparable model considered here is

$$\rho(\mathbf{x}, \mathbf{x}') = \left(|d_u|^{2\alpha} + 1\right)^{-1} \exp\left(-\frac{\sqrt{|h_1|^2/c_1^2 + |h_2|^2/c_2^2}}{\left(|d_u|^{2\alpha} + 1\right)^{\beta/2}}\right), \tag{4.19}$$

where $d_u = \sqrt{\sum_{i=1}^{k_\xi} \frac{u_i^2}{\phi_i^2}}, u_i = |\xi_i - \xi_i'|, h_j = |s_j - s_j'|$ for $i = 1, \ldots, k_\xi$ and $j = 1, 2$. A small nugget effect $\tau^2 = 10^{-6}$ was fixed during the parameter estimation step. We applied the FSA-Block approach with 200 knots selected by LHS and 100 blocks created by K-means algorithm to the nonseparable model, making computations of model implementation feasible. After the parameter estimation step, the posterior means of model parameters were plugged in (4.14) to make predictions at a finer $100 \times 64 \times 64$ input-spatial grid. We considered the separable model in (4.19) with $\beta = 0$ and the squared exponential model for comparisons. Although this problem has a high-dimensional input space (50 input variables), we experimented using the first $k_\xi = 3$ input variables (corresponding to the first 3 leading terms in the K-L expansion of $G(\cdot)$). We also experimented using a larger number of input variables ($k_\xi = 5$), but there are only slight differences for the prediction performances. So we focused on the smaller dimension $k_\xi = 3$ case.

Table 4.3 gives the parameter estimation results of different models as well as MSPEs. For the nonseparable model, the posterior mean estimate of β is close to 1, implying modeling the interaction effect may be beneficial. Figure 4.6 shows the predictive mean response surfaces in spatial domain by the nonseparable model, using the first 3 input variables. We can see the spatial patterns of the predictive mean response surfaces are very similar to the Monte Carlo estimates using the computer

Table 4.3: Posterior means of model parameters and prediction results of each output component.

$n_\xi = 24$	Nonsep	Sep ($\tau^2 = 0.01$)	Sqexp ($\tau^2 = 0.01$)
ϕ_1	5.564	9.689	25.722
ϕ_2	95.443	11.111	34.041
ϕ_3	142.046	47.540	76.473
c_1	1.265	0.686	0.724
c_2	1.258	0.222	0.420
α	0.083	0.0938	$-$
β	0.999	0	$-$
B_{10}	-0.478	0.1038	-0.177
B_{20}	-0.550	-0.0033	-0.498
B_{30}	-0.026	0.0002	0.003
Σ_{11}	0.0086	0.0099	10.758
Σ_{12}	0.0015	0.0016	2.946
Σ_{13}	2.8×10^{-5}	2.9×10^{-5}	0.0164
Σ_{22}	0.0084	0.0036	8.491
Σ_{23}	3.9×10^{-5}	2.6×10^{-5}	0.056
Σ_{33}	0.0001	0.0001	0.139
MSPE$_{sp}$	$(0.0035, 0.0036, 0.1030)$	$(0.0029, 0.0028, 0.1030)$	$(0.0033, 0.0035, 0.1030)$

model outputs, which can be viewed as the true values. Figure 4.7 gives the error bars of the predictive response surfaces in spatial domain.

For the separable models, we also used $k_\xi = 3$ in order to do fair comparisons with the nonseparable model. When we fixed a small nugget $\tau^2 = 10^{-6}$ for the input and spatial covariance parts, the estimates of parameters of the two separable models had relatively large variances, due to the numerical instability. Especially for the squared exponential model assuming infinite smoothness in input space, it yielded estimates of mean parameters with very large variances (over 10^6) and can not obtain reasonable prediction results. Then we increased $\tau^2 = 0.01$ suggested in [7] for these separable models and the results are shown in Table 4.3. The prediction results of the separable model with $\beta = 0$ are better than those of the nonseparable model, this may be because a covariance function separable in input and space was used in the log-normal process to model the permeability field K, which might lead to

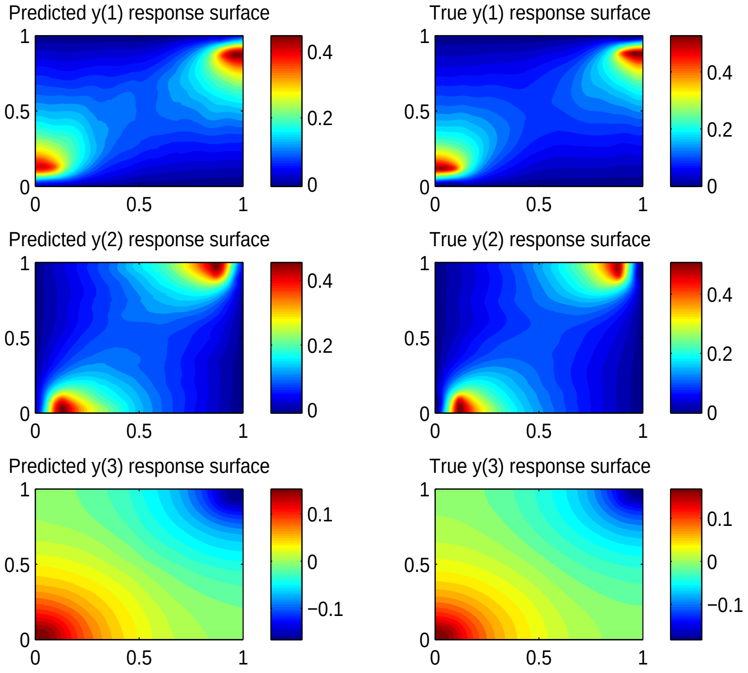
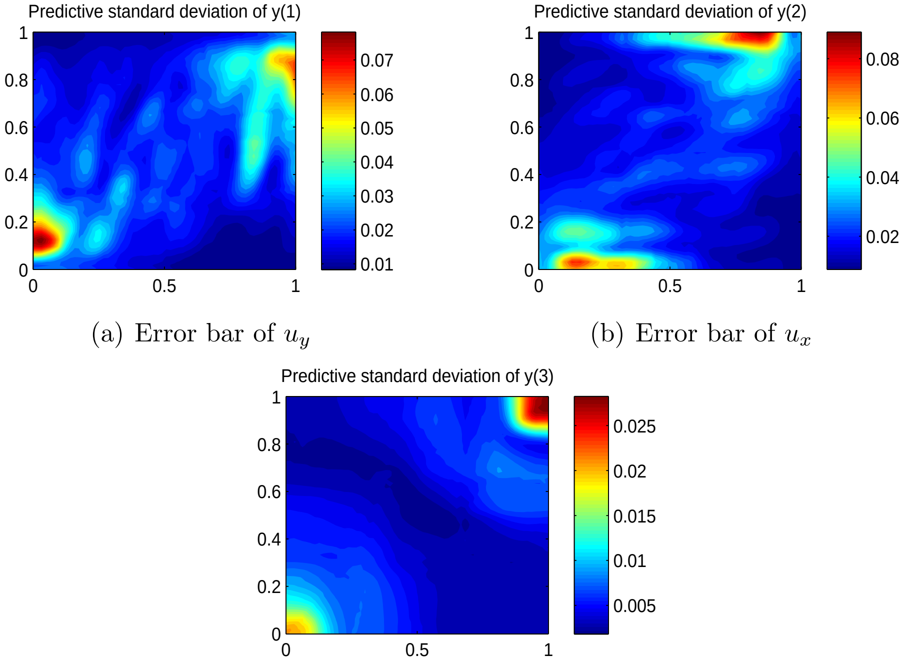


Figure 4.6: Predictive mean surfaces by the nonseparable model versus the Monte Carlo estimates based on 24 input points. Upper panels: velocity in y-direction u_y; middle panels: velocity in x-direction u_x; lower panels: pressure p.

certain level of separability in input and space for the outputs. Besides, the separable model in (4.19) with $\beta = 0$ is superior to the squared exponential model in terms of prediction, this may be due to an additional parameter α modeling the smoothness in input space. We remark that although the squared exponential separable model produced reasonable prediction results, the estimates of variance are in general fairly large compared with the data scales. In contrast, the nonseparable model with the FSA-Block approximation does not suffer much with the numerical stability problem, this may be because the correlations cross data blocks are approximated well by that of the reduced rank correlation model.



(a) Error bar of u_y

(b) Error bar of u_x

(c) Error bar of the pressure p

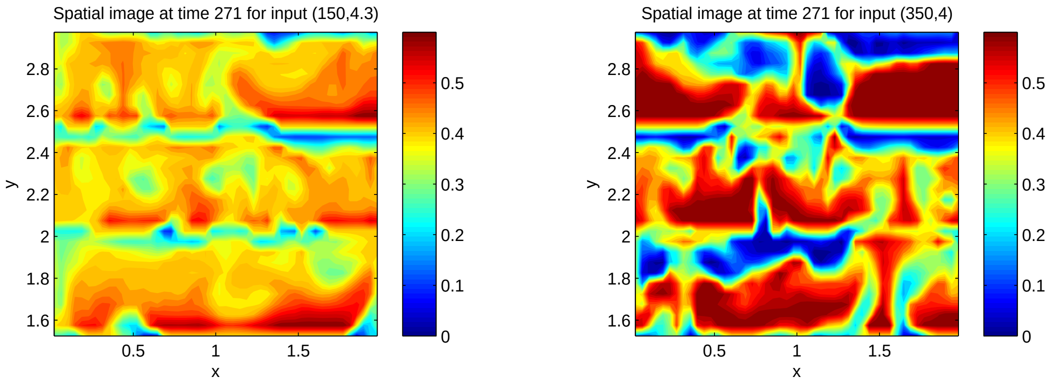
Figure 4.7: Predictive standard deviations by the nonseparable model.

4.5 The Regenerator of A Carbon Capture Unit

In this section, we apply the nonseparable Gaussian process surrogate model to a real example from the regenerator of a carbon capture unit. A carbon capture unit provides an alternative solution for limiting the carbon dioxide (CO_2) emissions. All carbon capture units contain an absorber device and a regenerator device. The solid sorbent particles capable of reacting with the CO_2 gas are looped through these two devices. In the absorber, the exhaust flue gas from a power plant reacts with the solid sorbent particles and its CO_2 component is trapped. Then after further processing steps, the cleaned exhaust flue gas is released into the atmosphere and the depleted sorbent particles are transferred to the regenerator. In the regenerator, the reverse chemical reaction is done to release CO_2 from the depleted sorbent particles for further processing (i.e. liquefaction and sequestration for long-term storage) and the regenerated sorbent particles are recycled back to the absorber. Since the bulk of the energy penalty is related to the regenerator, its efficiency is of crucial interests.

Recently, [61] developed a computational fluid dynamics (CFD)-based model for the fluid dynamics of the regenerator. The flow of sorbent particles is characterized by the density of solid volume fraction, which is sensitive to the system operating conditions such as the particle diameter d_p (with unit micro meters, μm) and the scaled velocity v_g/u_{mf} of gas injected at the bottom inlet (dimensionless; details see [27]), denoted by $(d_p, v_g/u_{mf})$. Figure 4.8 shows the solid volume fractions for 2 input points at a given time. It is clear that both the value and the spatial pattern of the solid volume fractions can change drastically for different input points. If the intermediate solid volume fraction values in $[0.2, 0.4]$ are more likely to result in better efficiency of the regenerator device, then the input point $(150, 4.3)$ is superior to $(350, 4)$ according to figure 4.8, since it has a larger proportion of intermediate solid volume fractions. Specifying the operation conditions in favor of a certain range of solid volume fractions needs a number of computer simulations. The CFD-based simulations are very time-consuming, taking days to complete one simulation under paralleled computing system. So it is challenging to run a large number of simulations to study the behaviors of the sorbent distribution under different operating conditions. Therefore, the Gaussian process model is used instead as an effective tool for the uncertainty quantification purpose [46].

The computer model outputs are the solid volume fractions over a time period, ranging from 0 to 0.6083. We focused on solid volume fraction values in $(0.1, 0.6]$, since without much knowledge of the reaction kinetics, we expect that the intermediate values are more likely to result in better performance of the regenerator [46]. We focused on the discrete distribution of the solid volume fractions created using 5



(a) $d_p = 150$ and $v_g/u_{mf} = 4.3$

(b) $d_p = 350$ and $v_g/u_{mf} = 4$

Figure 4.8: Images of the solid volume fractions at time 271 for 2 input points.

equal length bins $(0.1, 0.2], \cdots, (0.5, 0.6]$, aiming to check effects of the distribution of the solid volume fractions on the reaction kinetics. Denote the response vector by $\mathbf{f}(\boldsymbol{\xi}, t) = (\pi_1, \pi_2, \cdots, \pi_5)^T$, it was treated as a function of 2 input variables d_p and v_g/u_{mf}, as well as time t.

The training data set was on a 46×101 input-time grid, and we randomly held out 4 input points on the same time grid for evaluating model performances. The Gaussian process regression model in (4.1) with constant means was fitted to this data. We used the same nonseparable correlation function as in Section 4.4.2. The FSA approach with 15 blocks and 300 knots were applied to replace the full covariance model to speed up computations. The results of the same two separable models as in previous studies were again included for comparisons.

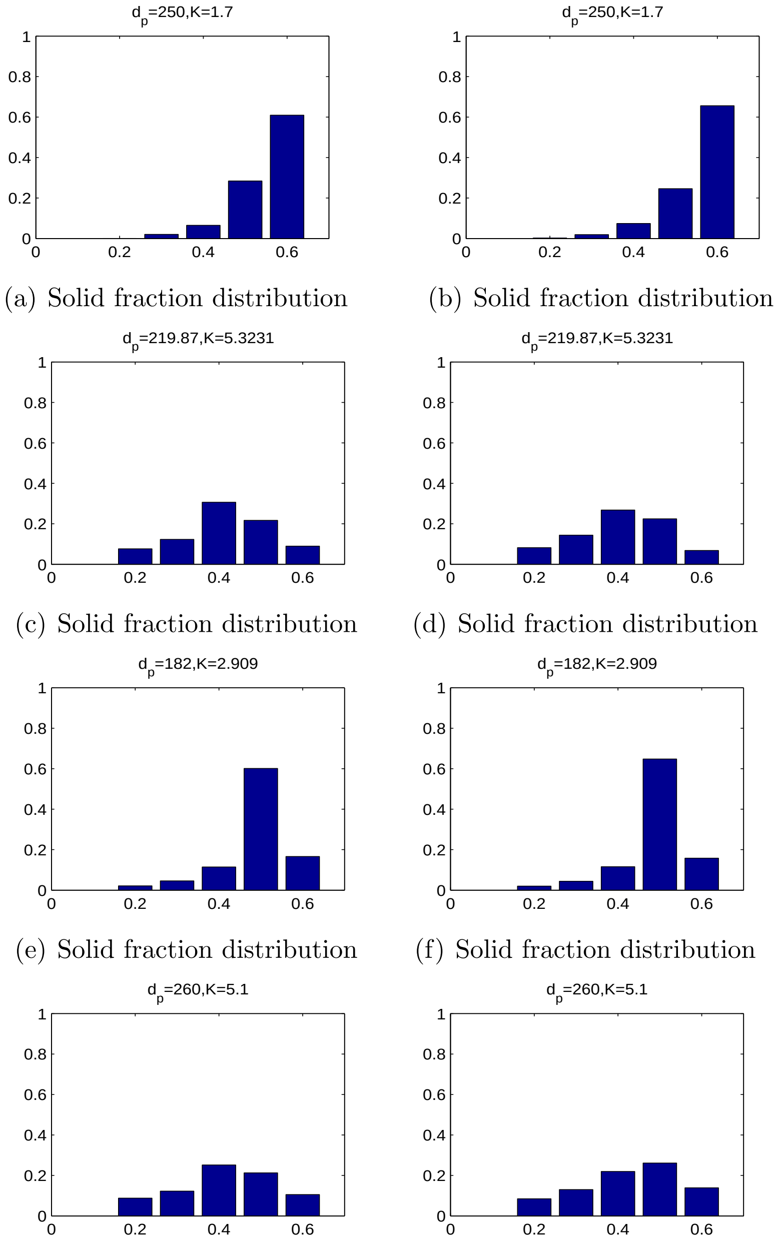
The parameter estimation and prediction results are summarized in Table 4.4. The posterior mean estimate of the input-time interaction parameter β of the nonseparable model is very close to 1, indicating the existence of the interaction effect. The nonseparable model and the separable model in (4.4) with $\beta = 0$ have close estimates of the smoothness parameter α, which is not surprising since it is related

Table 4.4: Posterior means of model parameters and the overall MSPEs.

$n = 4646$	a	c_1	c_2	α	β	B_{10}	B_{20}	B_{30}	B_{40}	B_{50}
Nonsep	15.131	92.993	0.943	0.638	0.993	0.095	0.120	0.138	0.178	0.236
Sep	6.263	485.10	6.791	0.586	0	0.104	0.134	0.087	0.110	0.252
Sqexp	1.147	499.280	7.349	–	–	0.105	0.136	0.086	0.106	0.257
	Σ_{11}	Σ_{12}	Σ_{13}	Σ_{14}	Σ_{15}	Σ_{22}	Σ_{23}	Σ_{24}	Σ_{25}	Σ_{33}
Nonsep	0.0020	−0.0003	−0.0001	0.0048	−0.0054	0.0038	−0.0001	0.0199	−0.0224	0.0037
Sep	0.0088	−0.0037	−0.0010	−0.0002	0	0.0119	−0.0061	−0.0008	0.0001	0.0172
Sqexp	0.0209	−0.0081	−0.0022	−0.0009	−0.0001	0.0277	−0.0132	−0.0021	−0.0004	0.0392
	Σ_{34}	Σ_{35}	Σ_{44}	Σ_{45}	Σ_{55}	MSPE				
Nonsep	0.0120	−0.0151	0.3009	−0.3310	0.3677	0.0011				
Sep	−0.0086	−0.0011	0.0233	−0.0140	0.0191	0.0017				
Sqexp	−0.0194	−0.0030	0.0615	−0.0401	0.0548	0.0018				

to the process properties in the dimension of time. Besides, all three correlation models produced a large estimate of the range parameter in the dimension of d_p and a small estimate of the range parameter in the dimension of v_g/u_{mf}, indicating that the computer code outputs are more sensitive to the v_g/u_{mf} variable.

In terms of prediction, the nonseparable model with the FSA approximation approach outperforms the two separable models. Specifically, the nonseparable model has the same prediction performance as the separable models for π_1, π_2, π_3, but it has smaller MSPEs $(0.0022, 0.0013)$ for π_4 and π_5, compared with $(0.0032, 0.0033)$ by the separable model in (4.4) with $\beta = 0$ and $(0.0034, 0.0036)$ by the squared exponential model. Figure 4.9 shows the predictive probabilities by the nonseparable model and the real computer code results for the hold-out set of 4 input points. We can observe that the predictive probabilities are close to the real data results in general. Figure 4.10 shows the predictive mean input surfaces of 5 probabilities by the nonseparable model, where the predictions were made on a dense 70×70 input grid at time points $t = 270, 271, \ldots, 280$. Based on these predictive mean surfaces of the input space, particular input regions can be found to improve the efficiency of the regenerator unit. For example, if the intermediate solid volume fractions in



(a) Solid fraction distribution

(b) Solid fraction distribution

(c) Solid fraction distribution

(d) Solid fraction distribution

(e) Solid fraction distribution

(f) Solid fraction distribution

(g) Solid fraction distribution

(h) Solid fraction distribution

Figure 4.9: The left panels are predictive distributions of solid volume fractions under different combinations of d_p and $K = v_g/u_{mf}$; the right panels are corresponding computer code results.

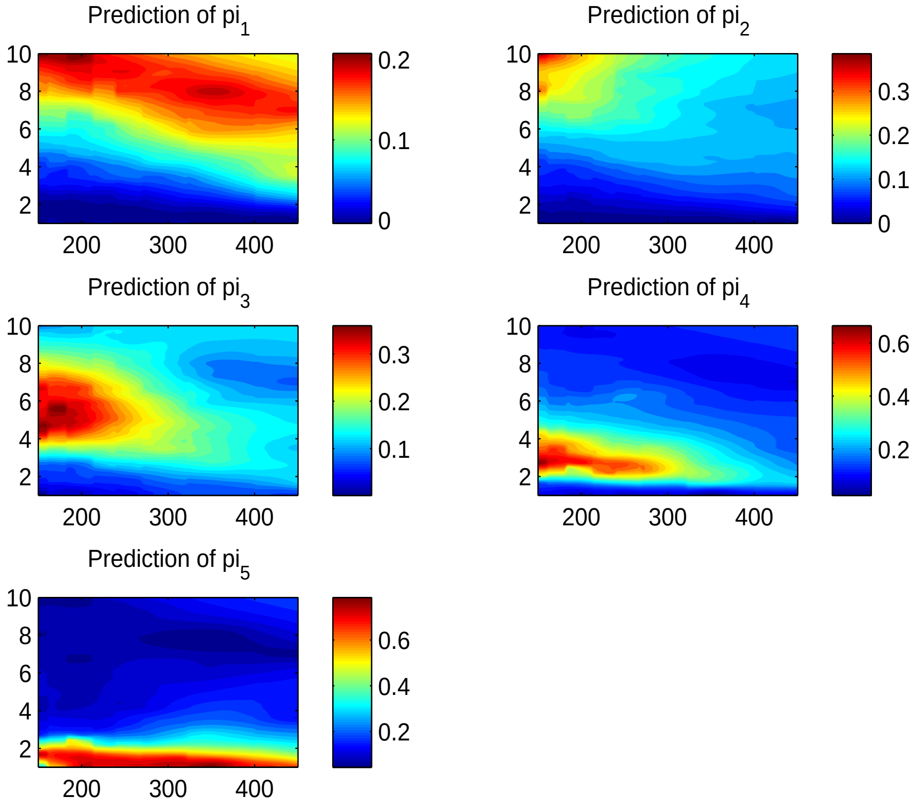


Figure 4.10: Predictive mean input surfaces of 5 probabilities by the nonseparable model.

$(0.3, 0.4]$ would result in better efficiency of the regenerator unit, then we may pay special attention to the high probability area of the predictive mean surface of π_3. For example, Figure 4.10 indicates that $d_p \in (150\mu m, 250\mu m)$ and $v_g/u_{mf} \in (4, 8)$ might be good choices of operating conditions. Other predictive mean surfaces of probabilities may also be useful for specifying the values of input variables that can result in good efficiency of the regenerator.

4.6 Discussion

In this Section, we extended the commonly used separable covariance Gaussian process surrogate models by using a more flexible nonseparable auto-covariance function, which includes separable model as a special case. The nonseparable model has the advantage of not only modeling dependence within each dimension in input, space and time but also interactions in dependence among different dimensions. This model relaxes the restrictions imposed by separable models on the conditional and marginal properties of a Gaussian process [55], and hence has broader applications especially in cases where the assumption of separability is problematic.

We also introduced a new computational method, referred to as the full-scale approximation with block modulating function approach (FSA-Block), to ease computational burdens associated with fitting the proposed nonseparable model to large computer code outputs. We illustrated the effectiveness of the nonseparable model with the FSA-Block approach through various simulation examples and a real data set from the computer code of the regenerator device of a carbon capture system.

The FSA-Block approach introduced here does not depend on the separable structure of the covariance matrix and hence can be used flexibly in various ways. For example, when the covariance function is partially separable and the number of observations in certain nonseparable dimensions is large, we can apply the FSA-Block approach only to the nonseparable part to facilitate computations. In this Section, we focused on the stationary auto-covariance functions. Nevertheless, this computational method also applies to nonstationary covariance functions such as the nonstationary model in [54], where spatial regions have different dependence structures.

5. CONCLUSIONS

This dissertation has discussed several approximation methods of Gaussian process models for large spatial and spatio-temporal datasets. Specifically, we have proposed a Smooth Full-Scale Approximation method for large spatial datasets, which extends the FSA-Block approach in the sense of preserving residual covariance among neighboring blocks. We also show that the proposed method can result in a valid Gaussian process so that both parameter estimation and prediction can be done in a unified framework. Since more residual correlations are preserved, the SFSA approach is less sensitive to the knot set. In addition, as a spatio-temporal extension of the FSA-Block approach, we have proposed a spatio-temporal full-scale approximations of covariance functions for large space-time datasets. Since the knot number and knot locations are crucial to the approximation performance, we introduce a Bayesian algorithm for automatically selecting the knot number and locations, to avoid the risk of the ad-hoc selection. Last, we have applied the FSA-Block approach to Gaussian process models for computer code outputs. To model the dependence structure of input, space and time, previous Gaussian process models use the separable covariance function for computational efficiency. We have proposed a multi-outputs Gaussian process model with nonseparable covariance functions to relax the separability. To facilitate computations, the FSA-Block approach is applied to approximate the nonseparable covariance model.

There are several potential extensions for the proposed methods in this dissertation. A natural extension of the proposed method in Section 2 is the spatio-temporal setting [42, 6, 78], where we can consider a spatio-temporal partition of responses and define the neighboring blocks in space and time. In this case, the Euclidean distance of spatio-temporal locations may not be a good measure of distance. We will also explore using other measures to define the block partition and neighboring blocks that minimize the residual covariance across non-neighboring blocks. In Section 2 and Section 3, we have used the K-means clustering algorithm to choose block centers and subsequently create the block partition, which is pre-specified for the proposed approximation method. An interesting direction for the future work is to treat the partition as unknown and select it adaptively using a Bayesian method, such as the tree-generating process [9, 31, 47].

In Section 4, we considered a separable cross-covariance structure among different outputs. A natural extension is to build a totally nonseparable emulator with both a nonseparable auto-covariance function and a nonseparable cross-covariance. The Linear Model of Coregionalization (LMC) [26, 25, 22] and the cross-covariance functions based on latent dimensions [1] may be applied to relax the separable cross-covariance assumption. Investigations on new computational methods are also problems of interest to facilitate the more demanding computational needs of the totally nonseparable model.

REFERENCES

[1] Tatiyana V Apanasovich and Marc G Genton. Cross-covariance functions for multivariate random fields based on latent dimensions. *Biometrika*, 97(1):15–30, 2010.

[2] K.E. Atkinson. *The numerical solution of integral equations of the second kind*, volume 4. Cambridge university press, 1997.

[3] Y. Bai, P. Song, and T. E. Raghunathan. Joint composite estimating functions in spatiotemporal models. *J. Roy. Statist. Soc. Ser. B*, 74(5):799–824, 2012.

[4] S. Banerjee, A.E. Gelfand, A.O. Finley, and H. Sang. Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(4):825–848, 2008.

[5] Sudipto Banerjee, Alan E Gelfand, and Bradley P Carlin. *Hierarchical modeling and analysis for spatial data*. In press, 2nd ed., 2014.

[6] M. Bevilacqua, C. Gaetan, J. Mateu, and E. Porcu. Estimating space and space-time covariance functions for large data sets: A weighted composite likelihood approach. *J. Amer. Statist. Assoc.*, 107:268–280, 2012.

[7] Ilias Bilionis, Nicholas Zabaras, Bledar A Konomi, and Guang Lin. Multi-output separable gaussian process: Towards an efficient, fully bayesian paradigm for uncertainty quantification. *J. Comput. Physics*, 241:212–239, 2013.

[8] Petruţa C Caragea and Richard L Smith. Asymptotic properties of computationally efficient alternative estimators for a class of multivariate normal models. *Journal of Multivariate Analysis*, 98(7):1417–1440, 2007.

[9] H. Chipman, E. George, and R. McCulloch. Bayesian CART model search. *Journal of the American Statistical Association*, 93:935–960, 1998.

[10] Stefano Conti and Anthony OHagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of statistical planning and inference*, 140(3):640–651, 2010.

[11] N. Cressie. *Statistics for Spatial Data*. Wiley, New York, 1993.

[12] N. Cressie and H.-C. Huang. Classes of nonseparable, spatio-temporal stationary covariance functions. *J. Amer. Statist. Assoc.*, 94:1330–1340, 1999.

[13] N. Cressie and G. Johannesson. Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):209–226, 2008.

[14] N. Cressie, T. Shi, and E. L. Kang. Fixed rank filtering for spatio-temporal data. *Journal of Computational and Graphical Statistics*, 19:724–745, 2010.

[15] Noel Cressie and Christopher K Wikle. *Statistics for spatio-temporal data*. John Wiley & Sons, 2011.

[16] Frank C Curriero and Subhash Lele. A composite likelihood approach to semivariogram estimation. *Journal of Agricultural, biological, and Environmental statistics*, pages 9–28, 1999.

[17] Carla Currin, Toby Mitchell, Max Morris, and Don Ylvisaker. Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association*, 86(416):953–963, 1991.

[18] Abhirup Datta, Sudipto Banerjee, Andrew O Finley, and Alan E Gelfand. Hierarchical nearest-neighbor gaussian process models for large geostatistical datasets. *arXiv preprint arXiv:1406.7343*, 2014.

[19] Jo Eidsvik, Benjamin A Shaby, Brian J Reich, Matthew Wheeler, and Jarad Niemi. Estimation and prediction in spatial models with block composite likelihoods. *Journal of Computational and Graphical Statistics*, (just-accepted), 2013.

[20] A. Finley, S. Banerjee, and A. Gelfand. Bayesian dynamic modeling for large space-time datasets using gaussian predictive processes. *J. Geogr. Syst.*, 14:29–47, 2012.

[21] A.O. Finley, H. Sang, S. Banerjee, and A.E. Gelfand. Improving the performance of predictive process modeling for large datasets. *Computational statistics & data analysis*, 53(8):2873–2884, 2009.

[22] Thomas E Fricker, Jeremy E Oakley, and Nathan M Urban. Multivariate gaussian process emulators with nonseparable covariance structures. *Technometrics*, 55(1):47–56, 2013.

[23] R. Furrer, M.G. Genton, and D. Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523, 2006.

[24] A. Gelfand, S. Banerjee, and D. Gamerman. Spatial process modelling for univariate and multivariate dynamic spatial data. *Environmetrics*, 16(5):465–479, 2005.

[25] Alan E Gelfand, Sudipto Banerjee, and Dani Gamerman. Spatial process modelling for univariate and multivariate dynamic spatial data. *Environmetrics*, 16(5):465–479, 2005.

[26] Alan E Gelfand, Alexandra M Schmidt, Sudipto Banerjee, and CF Sirmans. Nonstationary multivariate process modeling through spatially varying coregionalization. *Test*, 13(2):263–312, 2004.

[27] Dimitri Gidaspow. *Multiphase flow and fluidization: continuum and kinetic theory descriptions*. Academic press, 1994.

[28] E. Gilleland and D. Nychka. Statistical models for monitoring and regulating ground-level ozone. *Environmetrics*, 16:535–546, 2005.

[29] T. Gneiting. Correlation functions for atmospheric data analysis. *Quarterly Journal of the Royal Meteorological Society*, 125:2449–2464, 1999.

[30] T. Gneiting. Nonseparable, stationary covariance functions for space-time data. *J. Amer. Statist. Assoc.*, 97:590–600, 2002.

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

[32] Robert B Gramacy and Herbert KH Lee. Bayesian treed gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103(483), 2008.

[33] P.J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[34] R. Guhaniyogi, A. Finley, S. Banerjee, and A. Gelfand. Adaptive gaussian predictive process models for large spatial datasets. *Environmetrics*, 22:997–1007, 2011.

[35] William W Hager. Updating the inverse of a matrix. *SIAM review*, 31(2):221–239, 1989.

[36] W Keith Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[37] Dave Higdon. Space and space-time modeling using process convolutions. In *Quantitative methods for current environmental issues*, pages 37–56. Springer, 2002.

[38] Craig J Johns, Douglas Nychka, Timothy G F Kittel, and Chris Daly. Infilling sparse records of spatial fields. *Journal of the American Statistical Association*, 98(464):796–806, 2003.

[39] EE Kammann and Matthew P Wand. Geoadditive models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 52(1):1–18, 2003.

[40] Emily L Kang and Noel Cressie. Bayesian inference for the spatial random effects model. *Journal of the American Statistical Association*, 106(495):972–983, 2011.

[41] M. Katzfuss. Bayesian nonstationary spatial modeling for very large datasets. *Environmetrics*, 24(3):189–200, 2013.

[42] M. Katzfuss and N. Cressie. Spatio-temporal smoothing and em estimation for massive remote-sensing datasets. *J. Time Ser. Anal.*, 32:430–446, 2011.

[43] C. Kaufman, M. Schervish, and D. Nychka. Covariance tapering for likelihood-based estimation in large spatial data sets. *J. Amer. Statist. Assoc.*, 103(484):1545–1555, 2008.

[44] L. Kaufman and P. Rousseeuw. *Finding groups in data*, volume 16. Wiley, New York, 1990.

[45] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.

[46] Bledar Konomi, Georgios Karagiannis, Avik Sarkar, Xin Sun, and Guang Lin. Bayesian treed multivariate gaussian process with adaptive design: Application to a carbon capture unit. *Technometrics*, 56(2):145–158, 2014.

[47] Bledar A Konomi, Huiyan Sang, and Bani K Mallick. Adaptive bayesian non-stationary modeling for large spatial datasets using covariance approximations. *Journal of Computational and Graphical Statistics*, (just-accepted), 2013.

[48] Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between gaussian fields and gaussian markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011.

[49] Bruce G Lindsay. Composite likelihood methods. *Contemporary Mathematics*, 80(1):221–39, 1988.

[50] Kantilal Vardichand Mardia and Colin R Goodall. Spatial-temporal analysis of multivariate environmental monitoring data. *Multivariate Environmental S-tatistics*, 6(347-385):76, 1993.

[51] Bertil Matérn et al. Spatial variation. stochastic models and their application to some problems in forest surveys and other sampling investigations. *Meddelanden fran statens Skogsforskningsinstitut*, 49(5), 1960.

[52] M. D. McKay, W. J. Conover, and R. J. Beckman. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245, 1979.

[53] J. Oakley and A. O'Hagan. Bayesian inference for the uncertainty distribution of computer model outputs. *Biometrika*, 89(4):769–784, 2002.

[54] Christopher J Paciorek and Mark J Schervish. Spatial modelling using a new class of nonstationary covariance functions. *Environmetrics*, 17(5):483–506, 2006.

[55] Jonathan Rougier. A representation theorem for stochastic processes with separable covariance functions. Technical report, University of Bristol, 2011.

[56] H. Rue and H. Tjelmeland. Fitting gaussian markov random fields to gaussian fields. *Scandinavian Journal of Statistics*, 29(1):31–49, 2002.

[57] Håvard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications*, volume 104. Chapman & Hall, 2005.

[58] Jerome Sacks, William J Welch, Toby J Mitchell, Henry P Wynn, et al. Design and analysis of computer experiments. *Statistical science*, 4(4):409–423, 1989.

[59] H. Sang and J.Z. Huang. A full scale approximation of covariance functions for large spatial data sets. *Journal of the Royal Statistical Society Series B*, 74(1):111–132, 2012.

[60] H. Sang, M. Jun, and J.Z. Huang. Covariance approximation for large multivariate spatial data sets with an application to multiple climate model errors. *The Annals of Applied Statistics*, 5(4):2519–2548, 2011.

[61] Avik Sarkar, Wenxiao Pan, DongMyung Suh, E David Huckaby, and Xin Sun. Multiphase flow simulations of a moving fluidized bed regenerator in a carbon capture unit. *Powder Technology*, 2014.

[62] Michael L Stein. *Interpolation of spatial data: some theory for kriging*. Springer, 1999.

[63] Michael L Stein. Limitations on low rank approximations for covariance matrices of spatial data. *Spatial Statistics*, 8:1–19, 2014.

[64] Michael L Stein, Zhiyi Chi, and Leah J Welty. Approximating likelihoods for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(2):275–296, 2004.

[65] M.L. Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29:143–151, 1987.

[66] M.L. Stein. Space-time covariance functions. *J. Amer. Statist. Assoc.*, 100:310–321, 2005.

[67] Ying Sun, Bo Li, and Marc G Genton. Geostatistics for large datasets. In *Advances and challenges in space-time modelling of natural events*, pages 55–77. Springer, 2012.

[68] Luke Tierney. Markov chains for exploring posterior distributions. *the Annals of Statistics*, pages 1701–1728, 1994.

[69] Cristiano Varin, Gudmund Høst, and Øivind Skare. Pairwise likelihood inference in spatial generalized linear mixed models. *Computational statistics & data analysis*, 49(4):1173–1191, 2005.

[70] Cristiano Varin, Nancy Reid, and David Firth. An overview of composite likelihood methods. *Statistica Sinica*, 21(1):5–42, 2011.

[71] Cristiano Varin and Paolo Vidoni. A note on composite likelihood inference and model selection. *Biometrika*, 92(3):519–528, 2005.

[72] Aldo V Vecchia. Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 297–312, 1988.

[73] Jay M Ver Hoef, Noel Cressie, and Ronald Paul Barry. Flexible spatial models for kriging and cokriging using moving averages and the fast fourier transform (fft). *Journal of Computational and Graphical Statistics*, 13(2):265–282, 2004.

[74] Xiaoliang Wan and George Em Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, 209(2):617–642, 2005.

[75] H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in computational Mathematics*, 4(1):389–396, 1995.

[76] H. Wendland. Error estimates for interpolation by compactly supported radial basis functions of minimal degree 1. *Journal of Approximation Theory*, 93(2):258–272, 1998.

[77] Christopher K Wikle and Noel Cressie. A dimension-reduced approach to space-time kalman filtering. *Biometrika*, 86(4):815–829, 1999.

[78] Bohai Zhang, Huiyan Sang, and Jianhua Z Huang. Full-scale approximations of spatio-temporal covariance models for large datasets. *Statistica Sinica*, 25:99–114, 2015.

[79] Dale L Zimmerman. Computationally exploitable structure of covariance matrices and generalized convariance matrices in spatial models. *Journal of statistical computation and simulation*, 32(1-2):1–15, 1989.

APPENDIX A

PROOF OF THEOREMS

A.1 Proof of Theorem 2.2.1

Proof. Without loss of generality, let $\boldsymbol{\beta} = \mathbf{0}$ for notation convenience. We first prove that the approximated density in (2.4) is Gaussian. Let U denote $\mathcal{C}(S, S^*)\mathcal{C}_*^{-1}$, U_k denote $\mathcal{C}(S_k, S^*)\mathcal{C}_*^{-1}$, $\mathcal{S}_{N(k)}$ denote the location set of $\mathbf{Y}_{N(k)}$ and $U_{N(k)}$ denote $\mathcal{C}(S_{N(k)}, S^*)\mathcal{C}_*^{-1}$, then

$$\begin{aligned}
& \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \mathbf{w}^*, \boldsymbol{\theta}) \\
= \quad & (2\pi)^{-\frac{n}{2}} \cdot \prod_{k=1}^{K} |\Sigma_{con}^{(k)}|^{-\frac{1}{2}}| \cdot \exp\{-\frac{1}{2}\sum_{k=1}^{K}(\mathbf{Y}_k - U_k\mathbf{w}^* - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(\mathbf{Y}_{N(k)} - U_{N(k)}\mathbf{w}^*))^T \\
& \times\Sigma_{con}^{(k)-1}(\mathbf{Y}_k - U_k\mathbf{w}^* - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(\mathbf{Y}_{N(k)} - U_{N(k)}\mathbf{w}^*))\},
\end{aligned}$$

where $\Sigma_{con}^{(k)}$ is the residual conditional variance defined as $\Sigma_k - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}\Sigma_{k,N(k)}^T$, $\Sigma_k = \mathcal{C}_s(S_k, S_k) + \tau^2 I_{n_k}, \Sigma_{k,N(k)} = \mathcal{C}_s(S_k, S_{N(k)})$, and $\Sigma_{N(k)} = \mathcal{C}_s(S_{N(k)}, S_{N(k)}) + \tau^2 I_{n_{N(k)}}$. Next, we introduce some notations to obtain the quadratic term of the Gaussian density. Let

$$B_{kl} = \begin{cases} I_{n_k}, & \text{if } l = k; \\ -\Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}(, n_{(l-1)} + 1 : n_{(l)}), & \text{if } l \in N(k); \\ \mathbf{0}, & \text{otherwise,} \end{cases} \tag{A.1}$$

where $N(k)(i)$ is the i^{th} entry of $N(k)$, denoting the index of i^{th} neighboring block of block k, and $n_{(l)} = \sum_{i \geq 1, i <= l} n_{N(k)(i)}$. Let $B_k^* = (B_{k1}, \ldots, B_{kK})$, then $\mathbf{Y}_k - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}\mathbf{Y}_{N(k)} = B_k^*\mathbb{Y}$ and $U_k - \Sigma_{k,N(k)}\Sigma_{N(k)}^{-1}U_{N(k)} = B_k^*U$. Thus

$$\begin{aligned}
& \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \mathbf{w}^*, \boldsymbol{\theta}) \\
= \; & (2\pi)^{-\frac{n}{2}} \cdot \prod_{k=1}^{K} |\Sigma_{con}^{(k)}|^{-\frac{1}{2}}| \cdot \exp\{-\frac{1}{2}\sum_{k=1}^{K}(\mathbb{Y} - U\mathbf{w}^*)^T B_k^{*T}\Sigma_{con}^{(k)-1}B_k^*(\mathbb{Y} - U\mathbf{w}^*)\} \\
= \; & (2\pi)^{-\frac{n}{2}} \cdot |\Sigma_{con}|^{-\frac{1}{2}}| \cdot \exp\{-\frac{1}{2}(\mathbb{Y} - U\mathbf{w}^*)^T B^T\Sigma_{con}^{-1}B(\mathbb{Y} - U\mathbf{w}^*)\},
\end{aligned}$$

where $B_{n\times n} = (B_1^{*T}, B_2^{*T}, \ldots, B_K^{*T})^T$, and $\Sigma_{con} = \mathrm{diag}\{\Sigma_{con}^{(1)}, \Sigma_{con}^{(2)}, \ldots, \Sigma_{con}^{(K)}\}$. Since B_{kl} is a nonzero matrix only for $l \leq k$, by form of B, B is a $n \times n$ lower-triangular matrix with 1s as diagonal entries. Therefore $|B| = 1$ and it is clear that

$$\prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \mathbf{w}^*, \boldsymbol{\theta}) = \mathcal{N}(U\mathbf{w}^*, B^{-1}\Sigma_{con}B^{T^{-1}}). \tag{A.2}$$

The approximated data density

$$\begin{aligned}
& \tilde{p}(\mathbb{Y}|\boldsymbol{\theta}) \\
= \; & \int_{\mathbf{w}^*}\prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \mathbf{w}^*, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta})d\mathbf{w}^* \\
= \; & (2\pi)^{-\frac{n+m}{2}}\int_{\mathbf{w}^*}\exp\{-\frac{1}{2}(\mathbb{Y} - U\mathbf{w}^*)^T B^T\Sigma_{con}^{-1}B(\mathbb{Y} - U\mathbf{w}^*) - \frac{1}{2}\mathbf{w}^{*T}\mathcal{C}_*^{-1}\mathbf{w}^*\} \\
& \times|\mathcal{C}_*|^{-\frac{1}{2}} \cdot |\Sigma_{con}|^{-\frac{1}{2}}.
\end{aligned}$$

Note that $\Sigma_{\mathbf{w}^*}^{-1} = U^T B^T \Sigma_{con}^{-1} BU + \mathcal{C}_*^{-1}$ and $\boldsymbol{\mu}_{\mathbf{w}^*} = \Sigma_{\mathbf{w}^*} U^T B^T \Sigma_{con}^{-1} B\mathbb{Y}$, after integrating out $\mathbf{w}^*$

$$\begin{aligned}\tilde{p}(\mathbb{Y}|\boldsymbol{\theta}) &= (2\pi)^{-\frac{n}{2}} \exp\{-\frac{1}{2}\mathbb{Y}^T B^T (\Sigma_{con}^{-1} - \Sigma_{con}^{-1} BU\Sigma_{\mathbf{w}^*} U^T B^T \Sigma_{con}^{-1}) B\mathbb{Y}\} \\ &\quad \times |U^T B^T \Sigma_{con}^{-1} BU + \mathcal{C}_*^{-1}|^{-\frac{1}{2}} \cdot |\Sigma_{con}|^{-\frac{1}{2}} \cdot |\mathcal{C}_*|^{-\frac{1}{2}}\end{aligned}$$

By Sherman-Woodbury-Morrison inversion formula,

$$\Sigma_{con}^{-1} - \Sigma_{con}^{-1} BU\Sigma_{\mathbf{w}^*} U^T B^T \Sigma_{con}^{-1} = (\Sigma_{con} + BUC_* U^T B^T)^{-1}.$$

Using the fact that $|B| = 1$ and the Sylvester's theorem,

$$\begin{aligned}|(B^T(\Sigma_{con} + BUC^* U^T B^T)^{-1} B)^{-1}| &= |B^{-1}| \cdot |B^{T^{-1}}| \cdot |\Sigma_{con}| \cdot |I_n + \Sigma_{con}^{-1} BUC_* U^T B^T| \\ &= |\Sigma_{con}| \cdot |I_m + U^T B^T \Sigma_{con}^{-1} BUC_*| \\ &= |\Sigma_{con}| \cdot |C_*| \cdot |U^T B^T \Sigma_{con}^{-1} BU + C_*^{-1}|.\end{aligned}$$

So $\tilde{p}(\mathbb{Y}|\boldsymbol{\theta}) \sim \mathcal{N}(\mathbf{0}, B^{-1}\Sigma_{con} B^{T^{-1}} + U\mathcal{C}_* U^T)$.

□

A.2 Proof of Theorem 2.3.1

Proof. Without loss of generality, assume $\boldsymbol{\beta} = 0$. By assumption,

$$\begin{aligned}&\tilde{p}(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\theta}) \\ = &\int \tilde{p}(\mathbf{Y}_p|\mathbb{Y}, \mathbf{w}^*, \boldsymbol{\theta}) \cdot \tilde{p}(\mathbb{Y}|\mathbf{w}^*, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^* \\ = &\int \prod_{k=1}^{r} p(\mathbf{Y}_{p,k}|\mathbf{Y}_{M_k}, \mathbf{Y}_{N(M_k)}, \mathbf{w}^*, \boldsymbol{\theta}) \cdot \prod_{k=1}^{K} p(\mathbf{Y}_k|\mathbf{Y}_{N(k)}, \mathbf{w}^*, \boldsymbol{\theta}) \cdot p(\mathbf{w}^*|\boldsymbol{\theta}) d\mathbf{w}^*.\end{aligned}$$

Let $B_{p_k} = (B_{p_k,1}, \ldots, B_{p_k,K})$, where $B_{p_k,l}$ has the same definition as (2.5), then the quadratic term of $p(\mathbf{Y}_{p,k}|\mathbf{Y}_{M_k}, \mathbf{Y}_{N(M_k)}, \mathbf{w}^*, \boldsymbol{\theta})$ is

$$(\mathbf{Y}_{p,k} - U_{p_k}\mathbf{w}^* + B_{p_k}\mathbb{Y} - B_{p_k}U\mathbf{w}^*)^T \Sigma_{con}^{(p_k)^{-1}} (\mathbf{Y}_{p,k} - U_{p_k}\mathbf{w}^* + B_{p_k}\mathbb{Y} - B_{p_k}U\mathbf{w}^*).$$

Let $B_{p_k}^* = (\mathbf{0}, \ldots, I_{n_{p_k}}, \ldots, B_{p_k})$, $\tilde{\mathbf{Y}} = (\mathbf{Y}_{p,1}, \ldots, \mathbf{Y}_{p,r}, \mathbb{Y}^T)^T$, and $\tilde{U} = (U_{p_1}^T, \ldots, U_{p_r}^T, U^T)^T$, then

$$\begin{aligned}
\tilde{p}(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\theta}) \;\; \propto \;\; & \int \exp\{-\sum_{k=1}^{r} \frac{1}{2}(\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*)^T B_{p_k}^{*T} \Sigma_{con}^{(p_k)^{-1}} B_{p_k}^* (\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*)\} \\
& \times \exp\{-\frac{1}{2}(\mathbb{Y} - U\mathbf{w}^*)^T B^T \Sigma_{con}^{-1} B(\mathbb{Y} - U\mathbf{w}^*)\} \cdot \exp\{-\frac{1}{2}\mathbf{w}^* \mathcal{C}^{*-1}\mathbf{w}^*\} d\mathbf{w}^* \\
& \times \prod_{k=1}^{r} |\Sigma_{con}^{(p_k)}|^{-\frac{1}{2}} \cdot |\Sigma_{con}|^{-\frac{1}{2}} \cdot |\mathcal{C}^*|^{-\frac{1}{2}} d\mathbf{w}^*.
\end{aligned}$$

Let $B_p^* = (B_{p_1}^{*T}, \ldots, B_{p_r}^{*T})^T$ and $\Sigma_{con}^p = \mathrm{diag}\{\Sigma_{con}^{(p_1)}, \ldots, \Sigma_{con}^{(p_r)}\}$, then

$$\sum_{k=1}^{r} (\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*)^T B_{p_k}^{*T} \Sigma_{con}^{(p_k)^{-1}} B_{p_k}^* (\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*) = (\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*)^T B_p^{*T} \Sigma_{con}^{p^{-1}} B_p^* (\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*).$$

Let $B_p = (B_{p_1}^T, \ldots, B_{p_r}^T)^T$, $\tilde{B} = \begin{pmatrix} I_{n_p} & B_p \\ \mathbf{0} & B \end{pmatrix}$ and $\tilde{\Sigma}_{con} = \begin{pmatrix} \Sigma_{con}^p & \mathbf{0} \\ \mathbf{0} & \Sigma_{con} \end{pmatrix}$. Since $B_p^* = (I_{n_p},\ B_p)$, it can be shown that

$$\begin{aligned}
\tilde{p}(\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\theta}) \;\; \propto \;\; & \int \exp\{-\frac{1}{2}(\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*)^T \tilde{B}^T \tilde{\Sigma}_{con}^{-1} \tilde{B}(\tilde{\mathbf{Y}} - \tilde{U}\mathbf{w}^*) - \frac{1}{2}\mathbf{w}^* \mathcal{C}^{*-1}\mathbf{w}^*\} \\
& \times |\tilde{\Sigma}_{con}|^{-\frac{1}{2}} \cdot |\mathcal{C}^*|^{-\frac{1}{2}} d\mathbf{w}^*.
\end{aligned}$$

Similar to proof in Theorem 2.2.1, after integrating out $\mathbf{w}^*$,

$$\mathbf{Y}_p, \mathbb{Y}|\boldsymbol{\theta} \sim \mathcal{N}(\tilde{\mathbf{X}}\boldsymbol{\beta}, \tilde{B}^{-1}\tilde{\Sigma}_{con}\tilde{B}^{T^{-1}} + \tilde{U}\mathcal{C}_*\tilde{U}^T),$$

where $\tilde{\mathbf{X}} = (\mathbf{X}_p^T, \mathbf{X}^T)^T$. Since

$$\tilde{B}^{-1} = \begin{pmatrix} I_{n_p} & B_p \\ \mathbf{0} & B \end{pmatrix}^{-1} = \begin{pmatrix} I_{n_p} & -B_p B^{-1} \\ \mathbf{0} & B^{-1} \end{pmatrix},$$

by the conditional normal distribution fact, $\mathbf{Y}_p | \mathbb{Y}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}_p, \Sigma_p)$, where μ_p and Σ_p are defined in Theorem 2.3.1. □

APPENDIX B

CALCULATING THE APPROXIMATED $\pi(\boldsymbol{\theta}|Y)$ BY THE FSA-BLOCK APPROACH

The MCMC sampler requires evaluating $\log \pi(\boldsymbol{\theta}|Y)$ to obtain posterior samples of $\boldsymbol{\theta}$. By (4.13), we need to compute $|R|$, $|H^T R^{-1} H|$ and $|(Y - H\hat{B}_{gls})^T R^{-1}(Y - H\hat{B}_{gls})|$ to evaluate $\log \pi(\boldsymbol{\theta}|Y)$. Since

$$(Y - H\hat{B}_{gls})^T R^{-1}(Y - H\hat{B}_{gls}) = Y^T R^{-1} Y - Y^T R^{-1} H (H^T R^{-1} H)^{-1} H^T R^{-1} Y,$$

computing these determinants needs $H^T R^{-1} H$, $Y^T R^{-1} Y$ and $H^T R^{-1} Y$. When the sample size n is too large such that calculation of R^{-1} is prohibitive, we replace R^{-1} by $R^{\dagger -1}$ in (4.7) to make computations feasible. We give the implementation details below for evaluating the approximated posterior distribution of $\boldsymbol{\theta}$ by the FSA-Block approach.

Suppose $\cup_{i=1}^{K} B_i = \mathscr{X}$ is a partition of observed locations. Let Y_i denote the $n_i \times 1$ response vector in block i and H_i be the design matrix of Y_i such that $H = (H_1^T, \cdots, H_K^T)^T$. Since the approximated residual covariance R_ϵ is block-diagonal, let $R_\epsilon = \text{diag}(R_{\epsilon,1}, \cdots, R_{\epsilon,K})$ with i^{th} diagonal block $R_{\epsilon,i}$. Denote the cross-covariance between observations in B_i and the knot set $\mathscr{X}^*$ by $R_{n_i,n^*} = [\rho(\mathbf{x}, \mathbf{x}')]_{\mathbf{x} \in B_i, \mathbf{x}' \in \mathscr{X}^*}$ and the covariance of observations in B_i by $R_{n_i,n_i} = [\rho(\mathbf{x}, \mathbf{x}')]_{\mathbf{x}, \mathbf{x}' \in B_i}$. The approximated $\log \pi(\boldsymbol{\theta}|Y)$ by the FSA-Block approach can be evaluated in the following sequence:

1. Compute $R_{\epsilon,i} = R_{n_i,n_i} - R_{n_i,n^*} R_{**}^{-1} R_{n_i,n^*}^T$ for $i = 1, 2, \ldots, K$.

2. Calculate

$$
\begin{aligned}
Y^T R_\epsilon^{-1} Y &= \sum_{i=1}^{K} Y_i^T R_{\epsilon,i}^{-1} Y_i,\\
H^T R_\epsilon^{-1} H &= \sum_{i=1}^{K} H_i^T R_{\epsilon,i}^{-1} H_i,\\
H^T R_\epsilon^{-1} Y &= \sum_{i=1}^{K} H_i^T R_{\epsilon,i}^{-1} Y_i,\\
R_{nn^*}^T R_\epsilon^{-1} Y &= \sum_{i=1}^{K} R_{n_i,n^*}^T R_{\epsilon,i}^{-1} Y_i,\\
R_{nn^*}^T R_\epsilon^{-1} H &= \sum_{i=1}^{K} R_{n_i,n^*}^T R_{\epsilon,i}^{-1} H_i,\\
R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} &= \sum_{i=1}^{K} R_{n_i,n^*}^T R_{\epsilon,i}^{-1} R_{n_i,n^*}.
\end{aligned}
$$

3. Calculate

$$
\begin{aligned}
Y^T R^{\dagger-1} Y &= Y^T R_\epsilon^{-1} Y - Y^T R_\epsilon^{-1} R_{nn^*} (R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**})^{-1} R_{nn^*}^T R_\epsilon^{-1} Y,\\
H^T R^{\dagger-1} H &= H^T R_\epsilon^{-1} H - H^T R_\epsilon^{-1} R_{nn^*} (R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**})^{-1} R_{nn^*}^T R_\epsilon^{-1} H,\\
H^T R^{\dagger-1} Y &= H^T R_\epsilon^{-1} Y - H^T R_\epsilon^{-1} R_{nn^*} (R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**})^{-1} R_{nn^*}^T R_\epsilon^{-1} Y.
\end{aligned}
$$

4. Calculate the Cholesky decompositions of the following matrices

$$
\begin{aligned}
Y^T R^{\dagger-1} Y - Y^T R^{\dagger-1} H (H^T R^{\dagger-1} H)^{-1} H^T R^{\dagger-1} Y &= Q_1^T Q_1,\\
H^T R^{\dagger-1} H &= Q_2^T Q_2,\\
R_{nn^*}^T R_\epsilon^{-1} R_{nn^*} + R_{**} &= Q_3^T Q_3,\\
R_{\epsilon,i} &= Q_{\epsilon,i}^T Q_{\epsilon,i},\\
R_{**} &= Q_*^T Q_*.
\end{aligned}
$$

5. Compute the following log of determinants:

$$\log|Y^T R^{\dagger-1} Y - Y^T R^{\dagger-1} H (H^T R^{\dagger-1} H)^{-1} H^T R^{\dagger-1} Y| = 2\sum_{i=1}^{q} \log|Q_{1,ii}|,$$
$$\log|H^T R^{\dagger-1} H| = 2\sum_{i=1}^{m} \log|Q_{2,ii}|,$$
$$\log|R^{\dagger}| = 2\sum_{i=1}^{K}\sum_{j=1}^{n_i} \log|Q_{\epsilon,i,jj}| + 2\sum_{i=1}^{n^*} \log|Q_{3,ii}| - 2\sum_{i=1}^{n^*} \log|Q_{*,ii}|,$$

where $Q_{1,ii}$, $Q_{2,ii}$, $Q_{3,ii}$, $Q_{*,ii}$ are i^{th} diagonal entries of Q_1, Q_2, Q_3 and Q_* respectively; $Q_{\epsilon,i,jj}$ is j^{th} diagonal entry of $Q_{\epsilon,i}$ for $i = 1, 2, \ldots, K$.

6. Compute the approximated $\log \pi(\boldsymbol{\theta}|\boldsymbol{Y})$ by the FSA-Block method for sampling $\boldsymbol{\theta}$

$$\begin{aligned}\log \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{Y}) \;&=\; \log \pi(\boldsymbol{\theta}) - \frac{q}{2}\log|R^{\dagger}| - \frac{q}{2}\log|H^T R^{\dagger-1} H| + \log(C) \\ &\quad - \frac{n-m}{2}\log|Y^T R^{\dagger-1} Y - Y^T R^{\dagger-1} H (H^T R^{\dagger-1} H)^{-1} H^T R^{\dagger-1} Y|,\end{aligned}$$

where C is the normalizing constant of $\pi(\boldsymbol{\theta}|\boldsymbol{Y})$ and $\log(C)$ will be canceled when computing the acceptance ratio in the M-H algorithm.

Since $Y^T R^{\dagger-1} Y$, $H^T R^{\dagger-1} H$ and $(Y - H\hat{B}_{gls})^T R^{\dagger-1}(Y - H\hat{B}_{gls})$ have been calculated, we remark that (B, Σ) can also be efficiently sampled following the Algorithm 1.