Journal of Machine Learning Research 23 (2022) 1-32 Submitted 10/21; Revised 3/22; Published 4/22

# Kernel Packet: An Exact and Scalable Algorithm for Gaussian Process Regression with Matérn Correlations

**Haoyuan Chen** CHENHAOYUAN2018@TAMU.EDU
**Liang Ding** LDINGAA@TAMU.EDU
**Rui Tuo*** RUITUO@TAMU.EDU
*Wm Michael Barnes '64 Department of Industrial & Systems Engineering*
*Texas A&M University*
*College Station, TX 77843, USA*

**Editor:** Marc Peter Deisenroth

## Abstract

We develop an exact and scalable algorithm for one-dimensional Gaussian process regression with Matérn correlations whose smoothness parameter $\nu$ is a half-integer. The proposed algorithm only requires $\mathcal{O}(\nu^3 n)$ operations and $\mathcal{O}(\nu n)$ storage. This leads to a linear-cost solver since $\nu$ is chosen to be fixed and usually very small in most applications. The proposed method can be applied to multi-dimensional problems if a full grid or a sparse grid design is used. The proposed method is based on a novel theory for Matérn correlation functions. We find that a suitable rearrangement of these correlation functions can produce a compactly supported function, called a "kernel packet". Using a set of kernel packets as basis functions leads to a sparse representation of the covariance matrix that results in the proposed algorithm. Simulation studies show that the proposed algorithm, when applicable, is significantly superior to the existing alternatives in both the computational time and predictive accuracy.

**Keywords:** Computer experiments, Kriging, Uncertainty quantification, Compactly supported functions, Sparse matrices

## 1. Introduction

Gaussian process (GP) regression is a powerful function reconstruction tool. It has been widely used in computer experiments (Santner et al., 2003; Gramacy, 2020), spatial statistics (Cressie, 2015), supervised learning (Rasmussen, 2006), reinforcement learning (Deisenroth et al., 2013), probabilistic numerics (Hennig et al., 2015) and Bayesian optimization (Srinivas et al., 2009). GP regression models are flexible to fit a variety of functions, and they also enable uncertainty quantification for prediction by providing predictive distributions. With these appealing features, GP regression has become the primary surrogate model for computer experiments since popularized by Sacks et al. (1989). Despite these advantages, Gaussian process regression has its drawbacks. A major one is its computational complexity. Training a GP model requires furnishing matrix inverses and determinants. With $n$ training points, each of these matrix manipulations takes $\mathcal{O}(n^3)$ operations (referred to as "time" thereafter, assuming for simplicity that no parallel computing is enforced) if a direct method, such as the Cholesky decomposition, is

. First two authors contributed equally to this work.

applied. Besides, the computation for model training may also be hindered by the $\mathcal{O}(n^2)$ storage requirement (Gramacy, 2020) to store the $n \times n$ covariance matrix.

Tremendous efforts have been made in the literature to address the computational challenges of GP regression. Recent advances in scalable GP regression include random Fourier features (Rahimi and Recht, 2007), Nyström Approximation (also known as inducing points) (Smola and Schölkopf, 2000; Williams and Seeger, 2001; Titsias, 2009; Bui et al., 2017; Katzfuss, 2017; Chen and Stein, 2021), structured kernel interpolation Wilson and Nickisch (2015), etc. These methods are based on different types of approximation of GPs, i.e., the efficiency is gained at the cost of its accuracy. In contrast, the main objective of this work is to propose a novel scalable approach that does not need an approximation.

In this work, we focus on the use of GP regression in the context of *computer experiments*. In these studies, the training data are acquired through an experiment, in which the input points can be chosen. Such a choice is called a *design* of the experiment. It is well known that a suitably chosen design can largely simplify the computation. Here we consider the "tensor-space" techniques in terms of using a product correlation function and a full grid or a sparse grid (Plumlee, 2014) design. The tensor-space techniques can reduce a multivariate GP regression problem to several univariate problems. It is worth noting that, in some applications besides computer experiments, even if the input sites are not controllable, the data are naturally observed on full grids, e.g., the remote sensing data in geoscience applications (Bazi and Melgani, 2009). In these scenarios, the tensor-space techniques are also applicable.

Having the tensor-space techniques, the final hard nut to crack is the one-dimensional GP regression problem. We assume that the one-dimensional input data are already *ordered* throughout this work. This assumption is reasonable in computer experiment applications since the design points are chosen at our will. In other applications where we do not have ordered data in the first place, it takes only $\mathcal{O}(n \log n)$ time to sort them.

This work presents a mathematically *exact* algorithm to make conditional inference for one-dimensional GP regression with time and space complexity both linear in $n$. This algorithm is specialized for Matérn correlations with smoothness $\nu$ being a half-integer (see Section 1.1 for the definition.) Matérn correlations are commonly used in practice (Stein, 1999; Santner et al., 2003; Gramacy, 2020). In most applications, $\nu$ is chosen to be *small*, e.g., $\nu = 1.5$ or $\nu = 2.5$, for the sake of a higher model flexibility. The proposed algorithm enjoys the following important features.

- Given the hyper-parameters of the GP, the proposed algorithm is mathematically exact, i.e., all numerical error is attributed to the roundoff error given by the machine precision.
- There is no restriction for the one-dimensional input points. But if the points are equally spaced, the computational time can be further reduced.
- It takes only $\mathcal{O}(\nu^3 n)$ time to compute the matrix inversion and the determinant. For equally spaced designs, this time is further reduced to $\mathcal{O}(\nu^2 n)$.
- After the above pre-processing time, it takes only $\mathcal{O}(\nu + \log n)$ or even $\mathcal{O}(\nu)$ time to make a new prediction (i.e., evaluate the conditional mean) at an untried point.
- The storage requirement is only $\mathcal{O}(\nu n)$.

The remainder of this article is organized as follows. We will review the general idea of GP regression and some existing algorithms in Sections 1.1 and 1.2, respectively. The mathematical theory behind the proposed algorithm is introduced in Section 2. In Section 3, we propose the main algorithm. Numerical studies are given in Section 4. In Section 5, we briefly discuss some possible extensions of the proposed method. Concluding remarks are made in Section 6. Appendices A and B contain the required mathematical tools and our technical proofs, respectively.

### 1.1 A review on GP Regression

Let $Y(\mathbf{x})$ denote a stationary GP prior on $\mathbb{R}^d$ with mean function $\mu(\mathbf{x})$, variance $\sigma^2$, and correlation function $K(\mathbf{x}, \mathbf{x}')$. The correlation function is also referred to as a "kernel function" in the language of applied math or machine learning (Rasmussen, 2006). When $d = 1$, there are two types of popular correlation functions. The first type is the *Matérn* family (Stein, 1999):

$$K(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left( \sqrt{2\nu} \frac{|x - x'|}{\omega} \right)^{\nu} K_\nu \left( \sqrt{2\nu} \frac{|x - x'|}{\omega} \right), \tag{1}$$

for any $x, x' \in \mathbb{R}$, where $\nu > 0$ is the smoothness parameter, $\omega > 0$ is the scale and $K_\nu$ is the modified Bessel function of the second kind. The smoothness parameter $\nu$ governs the smoothness of the GP $Y$ (Santner et al., 2003; Stein, 1999); the scale parameter $\omega$ determines the spread of the correlation (Rasmussen, 2006). Matérn correlation functions are widely used because of its great flexibility. The second type is the *Gaussian* family:

$$K(x, x') = \exp\left( -\frac{|x - x'|^2}{\omega} \right), \tag{2}$$

for any $x, x' \in \mathbb{R}^d$. A Gaussian kernel function is the limit of a sequence of Matérn kernels with the smoothness parameter tending to infinity. The sample paths generated by GP with Gaussian correlation function are infinitely differentiable.

For multi-dimensional problems, a typical choice of the correlation structure is the *separable* or *product* correlation:

$$K(\mathbf{x}, \mathbf{x}') = \prod_{j=1}^{d} K_j(x_j, x_j'), \tag{3}$$

for any $\mathbf{x} = (x_1, \dots, x_d)^T, \mathbf{x}' = (x_1', \dots, x_d')^T$, where $K_j$ is a one-dimensional Matérn or Gaussian correlation function for each $j$. This assumption ensures that the GP lives in a tensor space, and is key to the "tensor-space" techniques, which reduces the multi-dimensional problems to one-dimensional ones.

Suppose that we have observed $\mathbf{Y} = \big(Y(\mathbf{x}_1), \cdots, Y(\mathbf{x}_n)\big)^T$ on $n$ distinct points $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$. The aim of *GP regression* is to predict the output at an untried input $\mathbf{x}^*$ by computing the distribution of $Y(\mathbf{x}^*)$ conditional on $\mathbf{Y}$, which is a normal distribution with the following conditional mean and variance (Santner et al., 2003; Banerjee et al., 2014):

$$\mathbb{E}\left[ Y(\mathbf{x}^*) \middle| \mathbf{Y} \right] = \mu(\mathbf{x}^*) + K(\mathbf{x}^*, \mathbf{X}) \mathbf{K}^{-1} (\mathbf{Y} - \boldsymbol{\mu}), \tag{4}$$

$$\mathrm{Var}\left[ Y(\mathbf{x}^*) \middle| \mathbf{Y} \right] = \sigma^2 \Big( K(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X}) \mathbf{K}^{-1} K(\mathbf{X}, \mathbf{x}^*) \Big), \tag{5}$$

where $\sigma^2 > 0$ is the variance of the stationary Gaussian process, $K(\mathbf{x}^*, \mathbf{X}) = (K(\mathbf{X}, \mathbf{x}^*))^T = (K(\mathbf{x}^*, \mathbf{x}_1), \cdots, K(\mathbf{x}^*, \mathbf{x}_n))$, $\mathbf{K} = [K(\mathbf{x}_i, \mathbf{x}_s)]_{i,s=1}^n$ and $\boldsymbol{\mu} = (\mu(\mathbf{x}_1), \cdots, \mu(\mathbf{x}_n))^T$.

In GP regression, the mean function $\mu$ is usually parametrized as a linear form $\mu = \sum_{i=1}^p \beta_i f_i$ for unknown coefficient vector $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p)^T$ and known regression functions $f_1, \cdots, f_p$. To improve the predictive performance of GP regression, the coefficient vector $\boldsymbol{\beta}$, variance $\sigma^2$ and scales $\boldsymbol{\omega} = (\omega_1, \cdots, \omega_d)^T$ associated to each one-dimensional correlation function $k_j$ are usually estimated via maximum likelihood (Jones et al., 1998). The log-likelihood function given the data is:

$$L(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}) = -\frac{1}{2}\left[n \log \sigma^2 + \log \det(\mathbf{K}) + \frac{1}{\sigma^2}(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{K}^{-1}(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})\right], \tag{6}$$

where $\det(\mathbf{K})$ denotes the determinant of the correlation matrix $\mathbf{K}$ and $\mathbf{F}$ is the $n \times p$ matrix whose $(i,s)^{\text{th}}$ entry is $f_s(\mathbf{x}_i)$. The *Maximum Likelihood Estimator* (MLE) is then defined as the maximimizer of the log-likelihood function: $(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \hat{\boldsymbol{\omega}}) = \operatorname{argmax}_{\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}} L(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega})$.

In both GP regression and parameter estimation, the computation can become unstable or even intractable because it involves the pursuit of the inversion and the determinant of the correlation matrix $\mathbf{K}$. Each task takes $\mathcal{O}(n^3)$ time if a direct method, such as the Cholesky decomposition, is applied, which is a fundamental computational challenge for GP regression.

### 1.2 Comparisons with Existing Methods

When applicable, as a specialized algorithm, the proposed method is significantly superior to the existing alternatives. In this section, we compare the proposed method with a few popular existing approaches for large-scale GP regression. It is worth noting that, the fundamental mathematical theory for the proposed method differs from that of any of the existing methods. A summary of the comparisons is presented in Table 1.

| Method | Kernels | Design | Time | Storage | Accuracy |
|---|---|---|---|---|---|
| Proposed Method | Matérn-$\nu$, $\nu - 1/2 \in \mathbb{N}$ | Arbitrary | $\mathcal{O}(\nu^3 n)$ | $\mathcal{O}(\nu n)$ | Exact |
| Toeplitz Methods | Stationary Kernels | Equally Spaced | $\mathcal{O}(n \log n)$ | $\mathcal{O}(n)$ | Depending on the number of iterations |
| Local Approximate GP (with $m$ nearest neighbors) | Arbitrary | Arbitrary | $\mathcal{O}(m^3)$ | $\mathcal{O}(m^2 + n)$ | Unknown |
| Random Fourier Features (with $m$ random features) | Matérn-$\nu$, $\nu > \frac{1}{2}$ Gaussian | Arbitrary | $\mathcal{O}(m^2 n)$ | $\mathcal{O}(m^2 + mn)$ | $\mathcal{O}_p(m^{-1/2})$ |
| Nyström Approximation (with $m$ inducing points) | Matérn-$\nu$, $\nu > \frac{3}{2}$ Gaussian | Arbitrary | $\mathcal{O}(m^2 n)$ | $\mathcal{O}(m^2 + mn)$ | Matérn: $\mathcal{O}_p(m^{-2\nu-1})$ Gaussian: $\mathcal{O}_p(\exp(-\alpha m \log m))$ |

Table 1: Comparisons with existing methods.

**Toeplitz methods**: Toeplitz methods (Wood and Chan, 1994) work for stationary GPs with *equally spaced* design points. These methods leverage the Toeplitz structure of the covariance matrices under this setting. To make a prediction in terms of solving (4) and (5), there are two approaches. The first is to solve the Toeplitz system exactly, using, for example, the Levinson algorithm (Zhang et al., 2005). This takes $\mathcal{O}(n^2)$ time. A more commonly used approach is based

on a conjugate gradient algorithm (Atkinson, 2008) to solve the matrix inversion problems in (4) and (5). Each step takes $\mathcal{O}(n \log n)$ time. For the sake of rapid computation, the number of iterations is chosen to be small. But then the method becomes inexact. Moreover, the conjugate gradient algorithm is unable to find the determinant in (6) (Wilson and Nickisch, 2015). Thus one has to resort to the exact algorithm to compute the likelihood value, which takes $\mathcal{O}(n^2)$ time. Toeplitz methods only works for equally spaced design points. This is a strong restriction for one-dimensional problems. For multi-dimensional problems in a tensor space, having this restriction can also be disturbing, especially under a sparse grid design. Many famous sparse grid designs are not based on equally spaced one-dimensional points, such as the Clenshaw-Curtis sparse grids (Gerstner and Griebel, 1998) or the ones suggested by Plumlee (2014).

**Local Approximate Gaussian Processes**: Gramacy and Apley (2015) proposed a sequential design scheme that dynamically defines the support of a Gaussian process predictor based on a local subset of the data. The local subset comprises of $m$ data points and, consequently, local approximate GP reduces the time and space complexity of GPs regression to $\mathcal{O}(m^3)$ and $\mathcal{O}(m^2+n)$ respectively. Local approximate GPs can achieve a decent accuracy level in empirical experiments but theoretical properties of this algorithm are still unknown.

**Random Fourier Features**: The class of Fourier features methods originates from the work by Rahimi and Recht (2007). These methods essentially use $\sum_{i=1}^{m} \phi_i(\mathbf{x})\phi_i(\mathbf{x}')$ to approximate $K(\mathbf{x}, \mathbf{x}')$, where $\phi_1(\mathbf{x}), \ldots, \psi_m(\mathbf{x})$ are basis functions constructed based on random samples from the spectral density, i.e., the Fourier transform of the kernel function $K$. This low-rank approximation reduces the time and space complexity of GP regression to $\mathcal{O}_p(m^2 n)$ and $\mathcal{O}_p(m^2+mn)$, respectively, with accuracy $\mathcal{O}_p(m^{-1/2})$ (Sriperumbudur and Szabo, 2015). Clearly, the price for fast computation of random Fourier features is the loss of its accuracy.

**Nyström Approximation**: These methods approximate the $n \times n$ covariance matrix $\mathbf{K}$ by an $m \times m$ matrix $\widetilde{\mathbf{K}} = K(\widetilde{\mathbf{X}}, \widetilde{\mathbf{X}})$, where $\widetilde{\mathbf{X}} = \{\tilde{\mathbf{x}}_i\}_{i=1}^m$ are called the inducing points. Similar to random Fourier features, Nyström approximations reduce the time complexity and space complexity of GP regression to $\mathcal{O}_p(m^2 n)$ and $\mathcal{O}_p(m^2 + mn)$, respectively. There are several approaches to choose the inducing points. Smola and Schölkopf (2000); Williams and Seeger (2001) selected $\widetilde{\mathbf{X}}$ from data points $\mathbf{X}$ by an orthogonalization procedure. Titsias (2009); Bui et al. (2017) treated $\widetilde{\mathbf{X}}$ as hidden variables and select these inducing points via variational Bayesian inference. Katzfuss (2017); Chen and Stein (2021) further developed the Nyström approximation to construct more precise kernel approximations with multi-resolution structures. For Matérn-$\nu$ kernel with $\nu > 3/2$, it is shown in Burt et al. (2019) that the accuracy level of any inducing points method is $\mathcal{O}_p(m^{-2\nu-1})$, which is higher than that of the random Fourier features. It was also shown in Tuo and Wang (2020) that GP regression with Matérn-$\nu$ kernel converges to the underlying true GP at the rate $\mathcal{O}(n^{-\nu})$ and, hence, the number of inducing points should satisfy $m = \mathcal{O}(n^{\frac{\nu}{2\nu+1}})$ to achieve the optimal order of approximation accuracy. In this case, the time and space complexity of Nyström approximations are $\mathcal{O}(n^{1+\frac{2\nu}{2\nu+1}})$ and $\mathcal{O}(n^{1+\frac{\nu}{2\nu+1}})$, respectively. These are higher than that of the proposed algorithm, not to mention that the latter provides the exact solutions.

**Other methods**: Using a compactly supported kernel (Gramacy, 2020) can induce a sparse covariance matrix, which can lead to an improvement in matrix manipulations. However, if the support of the kernel remains the same while the design points become dense in a finite interval, the sparsity of the covariance matrix is not high enough to improve the order of magnitude. On the other hand, shrinking the support may substantially change the sample path properties

of the GP, and impair the power of prediction. Recently, Loper et al. (2021) proposed a general approximation scheme for one-dimensional GP regression, which results in a linear-time inference method.

## 2. Theory of Kernel Packet Basis

In this section, we introduce the mathematical theory for the novel approach of inverting the correlation matrix in (4) and (5). Technical proofs of all theorems are deferred to Appendix B.

Direct inverting the matrix $\mathbf{K}$ in (4) and (5) is time consuming, because $\mathbf{K}$ is a dense matrix. Note that each entry of $\mathbf{K}$ is an evaluation of function $K(\cdot, x_j)$ for some $j$. The matrix $\mathbf{K}$ is not sparse because the support of $K$ is the entire real line. The main idea of this work is to find an *exact* representation of $\mathbf{K}$ in terms of sparse matrices. This exact representation is built in terms of a change-of-basis transformation.

In this section, we suppose $K$ is a one-dimensional kernel. Consider the linear space $\mathcal{K} = \text{span}\{K(\cdot, x_j)\}_{j=1}^n$. The goal is to find another basis for $\mathcal{K}$, denoted as $\{\phi_j\}_{j=1}^n$, satisfying the following properties:

1. Almost all of the $\phi_j$'s have *compact supports*.
2. $\{\phi_j\}_{j=1}^n$ can be obtained from $\{K(\cdot, x_j)\}_{j=1}^n$ via a *sparse linear transformation*, i.e., the matrix defining the linear transform from $\{K(\cdot, x_j)\}_{j=1}^n$ to $\{\phi_j\}_{j=1}^n$ is sparse.

Unless otherwise specified, throughout this article we assume that the one-dimensional kernel $K$ is a Matérn correlation function as in (1), whose spectral density is proportional to $\left(2\nu/\omega^2 + x^2\right)^{-(\nu+1/2)}$; see Rasmussen (2006); Tuo and Wu (2016). For notational simplicity, let $c^2 := 2\nu/\omega^2$ and the above spectral density is proportional to $(c^2 + x^2)^{-(\nu+1/2)}$.

### 2.1 Definition and Existence of Kernel Packets

In this section we introduce the theory that explains how we can find a compactly supported function in $\mathcal{K}$. Clearly, such a function must admit the representation $\phi(x) = \sum_{j=1}^n A_j K(x, x_j)$. Recall the requirement that the linear transform is sparse, which means that most of the coefficients $A_j$'s must be zero. This inspires the following definition.

**Definition 1** *Given a correlation function $K$ and input points $a_1 < \cdots < a_k$, a non-zero function $\phi$ is called a kernel packet (KP) of degree $k$, if it admits the representation $\phi(x) = \sum_{j=1}^k A_j K(x, a_j)$, and the support of $\phi$ is $[a_1, a_k]$.*

At first sight, it seems to be too optimistic to expect the existence of KPs. But, surprisingly, these functions do exist for one-dimensional Matérn correlation functions with half-integer smoothness. We will show that if the smoothness parameter $\nu$ is a half integer, i.e., $\nu - 1/2 \in \mathbb{N}$, there is a KP of degree $k := 2\nu + 2$ given any $k$ distinct input points.

For simplicity, we will use $k$ to parametrize the Matérn correlation, in other words, $\nu = (k-2)/2$ for $k = 3, 5, 7, \ldots$. Let $\mathbf{a} = (a_1, \ldots, a_k)^T$ be a vector with $a_1 < \cdots < a_k$. The goal is to find coefficients $A_j$'s such that

$$\phi_{\mathbf{a}}(x) := \sum_{j=1}^k A_j K(x, a_j) \tag{7}$$

is a KP. We will first find a necessary condition for $A_j$'s, and next we will prove that such a condition is also sufficient. We apply the Paley-Wiener theorem (see Lemma 15 in Appendix A and Stein and Shakarchi (2003)), which states that $\phi_{\mathbf{a}}(x)$ has a compact support only if the inverse Fourier transform of $\phi_{\mathbf{a}}$, denoted as $\tilde{\phi}_{\mathbf{a}}(x)$, can be extended to an entire function, i.e., a complex-valued function that is holomorphic on the whole complex plane. Let $i = \sqrt{-1}$. Our convention of inverse Fourier transform is $\tilde{f}(\xi) = (2\pi)^{-1/2} \int_{-\infty}^{\infty} f(x) e^{i\xi x} dx$. Direct calculations show

$$\tilde{\phi}_{\mathbf{a}}(x) \propto \left[ \sum_{j=1}^{k} A_j \exp\{i a_j x\} \right] (c^2 + x^2)^{-(k-1)/2}, x \in \mathbb{R}.$$

Clearly, the analytic continuation of this function (up to a constant) is

$$\tilde{\phi}_{\mathbf{a}}(z) \propto \left[ \sum_{j=1}^{k} A_j \exp\{i a_j z\} \right] (c^2 + z^2)^{-(k-1)/2} =: \gamma(z)(c^2 + z^2)^{-(k-1)/2},$$

and this function can be defined at any $z \in \mathbb{C} \setminus \{\pm ci\}$. Note that the function $(c^2 + z^2)^{-(k-1)/2}$ has poles at $z = \pm ci$, each with multiplicity $(k-1)/2$. According to Paley-Wiener theorem, we have to make $\tilde{\phi}_{\mathbf{a}}(z)$ an entire function, which implies that $\gamma(\pm ci) = 0$, each with multiplicity $(k-1)/2$ as well. This condition leads to a set of equations[1]:

$$\gamma^{(j)}(ci) = 0, \qquad \gamma^{(j)}(-ci) = 0,$$

for $j = 0, \dots, (k-3)/2$, where $\gamma^{(j)}$ denotes the $j$th derivative of $\gamma$. Clearly, there are $k-1$ equations, which can be rewritten as

$$\sum_{j=1}^{k} A_j a_j^l \exp\{\delta c a_j\} = 0, \tag{8}$$

with $l = 0, \dots, (k-3)/2$ and $\delta = \pm 1$, which is a $(k-1) \times k$ linear system. All solutions to this system are real-valued vectors because all coefficients are real.

Next we study the property of the linear system (8) and the corresponding $\phi_{\mathbf{a}}$. Theorem 2 states that $\phi_{\mathbf{a}}$ can be uniquely determined by (8) up to a multiplicative constant.

**Theorem 2** *If $a_1, \dots, a_k$ are distinct, the solution space of (8) is one-dimensional, i.e., there do not exist two linearly independent solutions to (8).*

Another important property of (8) is that its solution is not affected by a shift of $\mathbf{a}$. Define $\mathbf{a} + t = (a_1 + t, \dots, a_n + t)^T$.

**Theorem 3** *The solution space of (8), as a function of $\mathbf{a}$, is invariant under a shift transformation $T_t(\mathbf{a}) = \mathbf{a} + t$ for any $t \in \mathbb{R}$.*

**Remark 4** *Theorem 3 suggests that we can apply a shift on $\mathbf{a}$ without affecting the solution space. It is worth noting that, although the solution space does not change theoretically, the condition*

1. This statement is formalized as Lemma 20 in Appendix B.

*number of the linear system (8) may change, which may significantly affect the numerical accuracy. In order to enhance the numerical stability in solving (8), we suggest standardizing* $\mathbf{a}$ *using transformation* $T_t(\mathbf{a}) = \mathbf{a} + t$ *such that* $a_1 + t = -(a_n + t)$*, i.e.,* $t = -(a_1 + a_n)/2$*. The same standardization technique will be employed in the proof of Theorem 5.*

Theorem 5 confirms that any non-zero $\phi_{\mathbf{a}}$ is indeed a KP.

**Theorem 5** *The support of any non-zero function* $\phi_{\mathbf{a}}$ *defined by (7) and (8) is* $[a_1, a_k]$.

In other words, we have the following Corollary 6.

**Corollary 6** *Let K be a Matérn correlation with smoothness* $\nu$*. If* $\nu$ *is a half integer, then K admits a KP with degree* $2\nu + 2$*. In addition, given* $a_1 < \cdots < a_k$*, function* $\phi_{\mathbf{a}}$ *with the form (7) is a KP if and only if the coefficients* $A_j$*'s are given by a non-zero solution to (8).*

Figure 1 illustrates that the linear combination of 5 components $\{K(\cdot, a_j)\}_{j=1}^5$ provides a compactly supported KP corresponding to Matérn-3/2 correlation function.

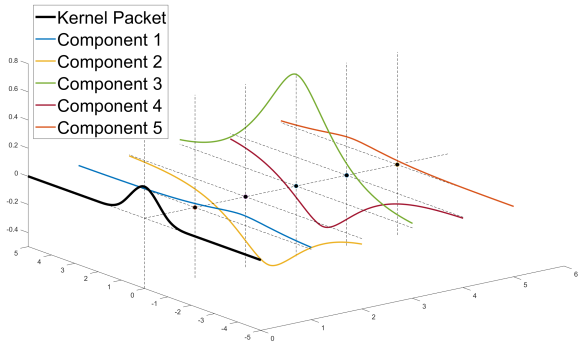


Figure 1: KP $\phi_{\mathbf{a}}$ (black line) corresponding to Matérn-3/2, and Matérn-3/2 correlation function and its components $\{K(\cdot, a_j)\}_{j=1}^5$.

It is evident that KPs are highly *non-trivial and precious*. Their existence relies on the correlation function. Theorem 7 shows that many other correlation function do not admit any KP, and consequently, the proposed algorithm is not applicable to these correlations.

**Theorem 7** *The following correlation functions do not admit KPs:*

1. *Any Matérn correlation function whose smoothness parameter is not a half integer.*
2. *Any Gaussian correlation function.*

Theorem 8 shows that the KP constructed by (8) has the lowest degree.

**Theorem 8** *Let K be a Matérn correlation function with half-integer smoothness* $\nu$*. Let m be a positive integer with* $m < 2\nu + 2$*. Then any function of the form* $\sum_{j=1}^m A_j K(\cdot, a_j)$ *does not have a compact support unless* $A_j = 0$ *for all* $j = 0, \ldots, m$*, and in other words, there does not exist a KP of degree lower than* $2\nu + 2$*.*

### 2.2 One-sided Kernel Packets

Besides KPs, we need to introduce a set of functions to capture the "boundary effects" of Gaussian process regression. As before, let $\mathbf{a} = (a_1, ..., a_s)^T$ be a vector with $a_1 < \cdots < a_s$. We consider the functions

$$\phi_{\mathbf{a}}(x) := \sum_{j=1}^{s} A_j K(x, a_j), \tag{9}$$

with $(k+1)/2 \le s \le k-1$ and a non-zero real vector $(A_1, \dots, A_s)^T$. Then Theorem 8 suggests that $\phi_{\mathbf{a}}$ in (9) cannot have a compact support. Nevertheless, it is possible that the support of $\phi_{\mathbf{a}}$ is a half real line. In this case, we cal $\phi_{\mathbf{a}}$ a one-sided KP. Specifically, we call $\phi_{\mathbf{a}}$ a *right-sided KP* if $\operatorname{supp} \phi_{\mathbf{a}} = [a_1, +\infty)$, and we call $\phi_{\mathbf{a}}$ a *left-sided KP* if $\operatorname{supp} \phi_{\mathbf{a}} = (-\infty, a_s]$.

First we consider right-sided KPs. We propose to identify $A_j$'s by solving

$$\sum_{j=1}^{s} A_j a_j^l \exp\{-c a_j\} = 0, \quad \sum_{j=1}^{s} A_j a_j^r \exp\{c a_j\} = 0, \tag{10}$$

where $l = 0, \dots, (k-3)/2$ and the second term of (10) comprises auxiliary equations for the case $s \ge (k+3)/2$ with $r = 0, \dots, s - (k+3)/2$. Similar to (8), (10) is an $(s-1) \times s$ linear system.

The following theorems describes the properties of the linear system (10) and the corresponding $\phi_{\mathbf{a}}$. Specifically, Theorem 11 confirms that $\phi_{\mathbf{a}}$ is indeed a right-sided KP.

**Theorem 9** *The solution space of* (10) *is one-dimensional provided that* $a_1, \dots, a_s$ *are distinct.*

**Theorem 10** *The solution space of* (10)*, as a function of* $\mathbf{a}$*, is invariant under a shift transformation* $T_t(\mathbf{a}) = \mathbf{a} + t$ *for any* $t \in \mathbb{R}$.

**Theorem 11** *The support of any non-zero function* $\phi_{\mathbf{a}}$ *defined by* (9) *and* (10) *is* $[a_1, +\infty)$.

Left-sided KPs are constructed similarly by solving the following equations:

$$\sum_{j=1}^{s} A_j a_j^l \exp\{c a_j\} = 0, \quad \sum_{j=1}^{s} A_j a_j^r \exp\{-c a_j\} = 0, \tag{11}$$

where $l = 0, \dots, (k-3)/2$ and the second term comprises auxiliary equations for the case $s \ge (k+3)/2$ with $r = 0, \dots, s - (k+3)/2$. The properties of left-sided KPs are analogous to these stated in Theorems 9-11, for which we omit the statements.

**Remark 12** *As in Remark 4, we suggest applying a shift transformation on* $\mathbf{a}$ *before computing* $A_j$*'s. Let* $T_t(\mathbf{a}) = (a_1', \dots, a_s')^T$*. We suggest using* $T_t$ *such that* $a_1' = 0$ *(i.e.,* $t = -a_1$*) for the right-sided KPs, and* $a_s' = 0$ *(i.e.,* $t = -a_s$*) for the left sided KPs. The same shifting is employed in the proof of Theorem 11.*

### 2.3 Kernel Packet Basis

Let $x_1 < \cdots < x_n$ be the input data, and $K$ a Matérn correlation function with a half-integer smoothness. Suppose $n \geq k$. We can construct the following $n$ functions, as a subset of $\mathcal{K}$:

1. $\phi_1, \phi_2, \ldots, \phi_{(k-1)/2}$, defined as left-sided KPs $\phi_{(x_1,\ldots,x_{(k+1)/2})}, \phi_{(x_1,\ldots,x_{(k+1)/2+1})}, \ldots, \phi_{(x_1,\ldots,x_{k-1})}$,
2. $\phi_{(k+1)/2}, \phi_{(k+1)/2+1}, \ldots, \phi_{n-(k-1)/2}$, defined as KPs $\phi_{(x_1,\ldots,x_k)}, \phi_{(x_2,\ldots,x_{k+1})}, \ldots, \phi_{(x_{n-k+1},\ldots,x_n)}$,
3. $\phi_{n-(k-3)/2}, \ldots, \phi_{n-1}, \phi_n$, defined as right-sided KPs $\phi_{(x_{n-k+2},\ldots,x_n)}, \ldots, \phi_{(x_{n-(k-1)/2-1},\ldots,x_n)}$, $\phi_{(x_{n-(k-1)/2},\ldots,x_n)}$.

Note that KPs and one-sided KPs given the input points cannot be uniquely defined. They are unique only up to a non-zero multiplicative factor. Here the choice of these factors are nonessential. The general theory and algorithms in this article will be valid for each specific choice. Now we present Theorem 13, which, together with the fact that the dimension of $\mathcal{K}$ is $n$, implies that $\{\phi_j\}_{j=1}^n$ forms a basis for $\mathcal{K}$, referred to as the *KP basis*.

**Theorem 13** *Let $x_1 < \cdots < x_n$ be the input data and the functions $\phi_1, \ldots, \phi_n$ are constructed in the above manner. Then the basis functions $\{\phi_j\}_{j=1}^n$ are linearly independent in $\mathcal{K}$.*

Further, it is straightforward to check via Theorems 5 and 11 that, given any $x \in \mathbb{R}$, the vector $\boldsymbol{\phi}(x) = (\phi_1(x), \ldots, \phi_n(x))^T$ has at most $k-1$ non-zero entries. As a result, we have constructed a basis for $\mathcal{K}$ satisfying the two sparse properties mentioned at the beginning of Section 2. Figure 2 illustrates a KP basis corresponding to Matérn-3/2 and Matérn-5/2 correlation function with input points $\mathbf{X} = \{0.1, 0.2, \ldots, 1\}$.

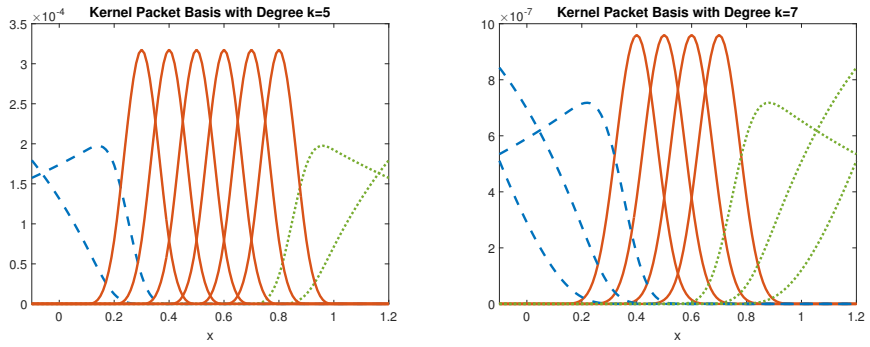


Figure 2: KP basis functions corresponding to Matérn-3/2 (left) and Matérn-5/2 (right) correlation function with input points $\mathbf{X} = \{0.1, 0.2, \ldots, 1\}$. The KPs, left-sided KPs, and the right-sided KPs are plotted in orange, blue, and green lines, respectively.

## 3. Kernel Packet Algorithms

In this section, we will employ the KP bases to develop scalable algorithms for GP regression problems. In Sections 3.1 and 3.2 , we present algorithms for one-dimensional GP regression with noiseless and noisy data, respectively. In section 3.3, we generalize the one-dimensional algorithms to higher dimensions by applying the tensor and sparse grid techniques.

### 3.1 One-dimensional GP Regression with Noiseless Data

The theory in Section 2 shows that for one-dimensional problems, given ordered and distinct inputs $x_1 < \cdots < x_n$, the correlation matrix $\mathbf{K}$ admits a sparse representation as

$$\mathbf{KA} = \boldsymbol{\phi}(\mathbf{X}), \tag{12}$$

where both $\mathbf{A}$ and $\boldsymbol{\phi}(\mathbf{X})$ are *banded matrices*. In (12), the $(l, j)^{\text{th}}$ entry of $\boldsymbol{\phi}(\mathbf{X})$ is $\phi_j(x_l)$. In view of the compact supportedness of $\phi_j$, $\boldsymbol{\phi}(\mathbf{X})$ is a banded matrix with bandwidth $(k-3)/2$:

$$\boldsymbol{\phi}(\mathbf{X}) = \begin{bmatrix} \ddots & & & & \\ \ddots & \phi_{j-\frac{k-3}{2}}(x_{j-2\frac{k-3}{2}}) & & & \\ \ddots & \vdots & \ddots & & \\ & \phi_{j-\frac{k-3}{2}}(x_j) & \cdots & \phi_{j+\frac{k-3}{2}}(x_j) & \\ & & \ddots & \vdots & \ddots \\ & & & \phi_{j+\frac{k-3}{2}}(x_{j+2\frac{k-3}{2}}) & \ddots \\ & & & & \ddots \end{bmatrix}.$$

The matrix of $\mathbf{A}$ consists of the coefficients to construct the KPs. In view of the sparse representation, $\mathbf{A}$ is a banded matrix with bandwidth $(k-1)/2$:

$$\mathbf{A} = \begin{bmatrix} \ddots & & & & \\ \ddots & A_{j-2\frac{k-1}{2},j-\frac{k-1}{2}} & & & \\ \ddots & \vdots & \ddots & & \\ & A_{j,j-\frac{k-1}{2}} & \cdots & A_{j,j+\frac{k-1}{2}} & \\ & & \ddots & \vdots & \ddots \\ & & & A_{j+2\frac{k-1}{2},j+\frac{k-1}{2}} & \ddots \\ & & & & \ddots \end{bmatrix}.$$

Computing $\mathbf{A}$ and $\boldsymbol{\phi}(\mathbf{X})$ takes $\mathcal{O}(k^3 n)$ time, because in the construction of each $\phi_j$, at most $k$ kernel basis functions are needed and the time complexity for solving the coefficients $\{A_{w,j} : |w - j| \leq \frac{k-1}{2}\}$ satisfying equation (8), (10) or (11) is $\mathcal{O}(k^3)$. The computational time $\mathcal{O}(k^3 n)$ in this step will dominate that in the next step, which is $\mathcal{O}(k^2 n)$. However, if the design points are equally spaces, the KP coefficients given by (8) will remain the same for each $k$ consecutive data points, so that we only need to compute these values once, and thus the computational time of this step is only $\mathcal{O}(k^4)$. In this case, the computation time in the next step, i.e., $\mathcal{O}(k^2 n)$, will be dominant, provided that $k \ll n$.

Now we solve the GP regression problem, by substituting the identity $K(\cdot, \mathbf{X}) = \boldsymbol{\phi}(\cdot)\mathbf{A}^{-1}$ into (4) and (5) to obtain

$$\mathbb{E}\left[Y(x^*)\middle|\mathbf{Y}\right] = \mu(x^*) + \boldsymbol{\phi}^T(x^*)\left[\boldsymbol{\phi}(\mathbf{X})\right]^{-1}(\mathbf{Y} - \boldsymbol{\mu}), \tag{13}$$

$$\mathrm{Var}\left[Y(x^*)\middle|\mathbf{Y}\right] = \sigma^2\Big(K(x^*, x^*) - \boldsymbol{\phi}^T(x^*)\left[\boldsymbol{\phi}(\mathbf{X})\right]^{-1} K(\mathbf{X}, x^*)\Big). \tag{14}$$

The key to GP regression now becomes calculating the vector $[\boldsymbol{\phi}(\mathbf{X})]^{-1}\mathbf{v}$ with $\mathbf{v} = \mathbf{Y} - \boldsymbol{\mu}$ or $\mathbf{v} = K(\mathbf{X}, x^*)$. This is equivalent to solving the sparse banded linear system $\boldsymbol{\phi}(\mathbf{X})\mathbf{s} = \mathbf{v}$. There exists quite a few sparse linear solvers that can solve this linear system efficiently. For example, the algorithm based on the LU decomposition in Davis (2006) can be applied to solve for $\mathbf{s}$ in $\mathcal{O}(k^2 n)$ time. MATLAB provides convenient and efficient builtin functions, such as `mldivide` or `decomposition`, to solve sparse banded linear system in this form.

It is worth noting that (13) can be executed in the following faster way when we need to evaluate $\mathbb{E}[Y(x^*)|\mathbf{Y}]$ for a many different $x^*$. First, we compute $\mathbf{s} := [\boldsymbol{\phi}(\mathbf{X})]^{-1}(\mathbf{Y} - \boldsymbol{\mu})$, which takes $\mathcal{O}(k^2 n)$ time. Next we evaluate $\mu(x^*) + \boldsymbol{\phi}^T(x^*)\mathbf{s}$ for different $x^*$. As said before, $\boldsymbol{\phi}^T(x^*)$ has at most $k-1$ non-zero entries; see Figure 2. If we know which $k-1$ entries are non-zero, the second step takes only $\mathcal{O}(k)$ time. To find the non-zero entry, a general approach is to use a binary search, which takes $\mathcal{O}(\log n)$ time. Sometime, these entries can be found within a constant time. For example, if the design points are equally spaced, there exist explicit expressions for the indices of the non-zero entries; if we need to predict for $x^*$ over a dense mesh (which is a typical task of surrogate modeling), we can use the indices of the non-zero entries for the previous point as an initial guess to find those for the current point.

Similar to the conditional inference, the log-likelihood function (6) can also be computed in $\mathcal{O}(k^2 n)$ time. First, the log-determinant of $\mathbf{K}$ can be rewritten as $\log\det(\mathbf{K}) = \log\det(\boldsymbol{\phi}(\mathbf{X})) - \log\det(\mathbf{A})$, according to identity (12): Because both $\mathbf{A}$ and $\boldsymbol{\phi}(\mathbf{X})$ are banded matrices, their determinants can be computed in $\mathcal{O}(k^2 n)$ time by sequential methods (Kamgnia and Nguenang, 2014, section 4.1). Second, the same method for the conditional inference can be applied to compute $(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{K}^{-1}(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})$ in $\mathcal{O}(k^2 n)$ time.

### 3.2 One-dimensional GP Regression with Noisy Data

Suppose we observe data $\mathbf{Z}$, which is a noisy version of $\mathbf{Y}$. Specifically, $Z(\mathbf{x}_i) = Y(\mathbf{x}_i) + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma_Y^2)$. In this case, the covariance of the observed noisy responses is $\mathrm{Cov}\big(Z(\mathbf{x}_i), Z(\mathbf{x}_j)\big) = \sigma^2 K(\mathbf{x}_i, \mathbf{x}_j) + \sigma_Y^2 \mathbb{I}(\mathbf{x}_i = \mathbf{x}_j)$. In other words, the covariance matrix $\mathrm{Cov}(\mathbf{Z}, \mathbf{Z})$ is $\sigma^2\mathbf{K} + \sigma_Y^2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. The posterior predictor at a new point $\mathbf{x}^*$ is also normal distributed with the following conditional mean and variance:

$$\mathbb{E}\left[Y(\mathbf{x}^*)\middle|\mathbf{Z}\right] = \mu(\mathbf{x}^*) + K(\mathbf{x}^*, \mathbf{X})\left[\mathbf{K} + \frac{\sigma_Y^2}{\sigma^2}\mathbf{I}\right]^{-1}(\mathbf{Z} - \boldsymbol{\mu}), \tag{15}$$

$$\mathrm{Var}\left[Y(\mathbf{x}^*)\middle|\mathbf{Z}\right] = \sigma^2\left(K(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X})\left[\mathbf{K} + \frac{\sigma_Y^2}{\sigma^2}\mathbf{I}\right]^{-1} K(\mathbf{X}, \mathbf{x}^*)\right), \tag{16}$$

and the log-likelihood function given data $\mathbf{Z}$ is:

$$L(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}) = -\frac{1}{2}\left[\log\det(\sigma^2\mathbf{K} + \sigma_Y^2\mathbf{I}) + \big(\mathbf{Z} - \mathbf{F}\boldsymbol{\beta}\big)^T\big[\sigma^2\mathbf{K} + \sigma_Y^2\mathbf{I}\big]^{-1}\big(\mathbf{Z} - \mathbf{F}\boldsymbol{\beta}\big)\right]. \tag{17}$$

When the input $\mathbf{x}$ is one dimensional, (15), (16), and (17) can be calculated in $\mathcal{O}(k^2 n)$ as the noiseless case because the covariance matrix $\sigma^2\mathbf{K} + \sigma_Y^2\mathbf{I}$ admits the following factorization:

$$\sigma^2\mathbf{K} + \sigma_Y^2\mathbf{I} = \big(\sigma^2\boldsymbol{\phi}(\mathbf{X}) + \sigma_Y^2\mathbf{A}\big)\mathbf{A}^{-1}. \tag{18}$$

By substituting (18) and the identity $K(\cdot,\mathbf{X}) = \boldsymbol{\phi}(\cdot)\mathbf{A}^{-1}$ into (15), (16), and (17), we can obtain:

$$\mathbb{E}\left[Y(x^*)\middle|\mathbf{Z}\right] = \mu(x^*) + \boldsymbol{\phi}^T(x^*)\left[\boldsymbol{\phi}(\mathbf{X}) + \frac{\sigma_Y^2}{\sigma^2}\mathbf{A}\right]^{-1}(\mathbf{Z}-\boldsymbol{\mu}), \tag{19}$$

$$\mathrm{Var}\left[Y(x^*)\middle|\mathbf{Z}\right] = \sigma^2\left(K(x^*,x^*) - \boldsymbol{\phi}^T(x^*)\left[\boldsymbol{\phi}(\mathbf{X}) + \frac{\sigma_Y^2}{\sigma^2}\mathbf{A}\right]^{-1}K(\mathbf{X},x^*)\right). \tag{20}$$

and

$$\begin{aligned} L(\boldsymbol{\beta},\sigma^2,\boldsymbol{\omega}) = -\frac{1}{2}\Big[ &\log\det\left(\sigma^2\boldsymbol{\phi}(\mathbf{X}) + \sigma_Y^2\mathbf{A}\right) - \log\det(\mathbf{A}) \\ &+ \left(\mathbf{Z}-\mathbf{F}\boldsymbol{\beta}\right)^T\mathbf{A}\left[\sigma^2\boldsymbol{\phi}(\mathbf{X}) + \sigma_Y^2\mathbf{A}\right]^{-1}\left(\mathbf{Z}-\mathbf{F}\boldsymbol{\beta}\right)\Big]. \end{aligned} \tag{21}$$

We have shown that $\boldsymbol{\phi}(\mathbf{X})$ and $\mathbf{A}$ are banded matrices with bandwidth $(k-3)/2$ and $(k-1)/2$, respectively. Therefore, the matrix $\sigma^2\boldsymbol{\phi}(\mathbf{X})+\sigma_Y^2\mathbf{A}$ is also a banded matrix with bandwidth $(k-3)/2$. Time complexity for computing this sum is $\mathcal{O}(kn)$. We then can use the algorithms for banded matrices introduced in section 3.1 to compute (19), (20), and (21) in time complexity $\mathcal{O}(k^2n)$. Recall that the time complexities for computing $\boldsymbol{\phi}(\mathbf{X})$ and $\mathbf{A}$ are both $\mathcal{O}(k^3n)$. Therefore, in the noisy setting, the total time complexity for computing the posterior and MLE is still $\mathcal{O}(k^3n)$, which is the same as the noiseless case.

### 3.3 Multi-dimensional KP

When data is noiseless, the exact algorithm proposed in Section 3.1 can be used to solve multi-dimensional problems if the input points are full or sparse grids.

A full grid is defined as the *Cartesian product* of one dimensional point sets: $\mathbf{X}^{\mathrm{FG}} = \times_{j=1}^{d}\mathbf{X}^{(j)}$ where each $\mathbf{X}^{(j)}$ denotes any one-dimensional point set. Assuming a separable correlation function (3) comprising $d$ one-dimensional Matérn correlation functions with half-integer smoothness, and inputs on a full grid $\mathbf{X}^{\mathrm{FG}}$, the covariance vector $K(\mathbf{x}^*,\mathbf{X}^{\mathrm{FG}})$ and covariance matrix $\mathbf{K}$ decompose into *Kronecker products* of matrices over each input dimension (Saatçi, 2012; Wilson, 2014):

$$K(\mathbf{x}^*,\mathbf{X}^{\mathrm{FG}}) = \bigotimes_{k=1}^{d}K_j(x_j^*,\mathbf{X}^{(j)}) = \bigotimes_{j=1}^{d}\boldsymbol{\phi}_j^T(x_j^*)\mathbf{A}_j^{-1} = \left(\bigotimes_{j=1}^{d}\boldsymbol{\phi}_j^T(x_j^*)\right)\left(\bigotimes_{j=1}^{d}\mathbf{A}_j^{-1}\right) \tag{22}$$

$$\mathbf{K} = \bigotimes_{k=1}^{d}K_j(\mathbf{X}^{(j)},\mathbf{X}^{(j)}) = \bigotimes_{j=1}^{d}\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\mathbf{A}_j^{-1} = \left(\bigotimes_{j=1}^{d}\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right)\left(\bigotimes_{j=1}^{d}\mathbf{A}_j^{-1}\right). \tag{23}$$

When we compute the vector $K(\mathbf{x}^*,\mathbf{X})\mathbf{K}^{-1}$, the matrix $\bigotimes_{j=1}^{d}\mathbf{A}_j^{-1}$ is cancelled as the one dimensional case. Therefore, (4) and (5) can be expressed as

$$\mathbb{E}\left[Y(\mathbf{x}^*)\middle|\mathbf{Y}\right] = \mu(\mathbf{x}^*) + \left(\bigotimes_{j=1}^{d}\boldsymbol{\phi}_j^T(x_j^*)\right)\left(\bigotimes_{j=1}^{d}\left[\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right]^{-1}\right)(\mathbf{Y}-\boldsymbol{\mu}) \tag{24}$$

$$\mathrm{Var}\left[Y(\mathbf{x}^*)\middle|\mathbf{Y}\right] = \sigma^2\left(K(\mathbf{x}^*,\mathbf{x}^*) - \prod_{j=1}^{d} \boldsymbol{\phi}_j^T(x_j^*)\left[\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right]^{-1} K_j(\mathbf{X}^{(j)}, x_j^*)\right) \tag{25}$$

and the log-likelihood function (6) becomes

$$\begin{aligned} L(\boldsymbol{\beta},\sigma^2,\boldsymbol{\omega}) = -\frac{1}{2}\Bigg[n\log\sigma^2 + \sum_{j=1}^{d}\frac{n}{n_j}\left(\log\det\boldsymbol{\phi}_j(\mathbf{X}^{(j)}) - \log\det\mathbf{A}_j\right) \\ + \frac{1}{\sigma^2}(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})^T\left(\bigotimes_{j=1}^{d}\mathbf{A}_j\right)\left(\bigotimes_{j=1}^{d}\left[\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right]^{-1}\right)(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})\Bigg], \end{aligned} \tag{26}$$

where $\boldsymbol{\phi}_j$'s are the KPs associated to correlation function $K_j$ and point set $\mathbf{X}^{(j)}$, $\mathbf{A}_j$ is the coefficient matrix for constructing $\boldsymbol{\phi}_j$ defined in (12), and $n = \prod_{j=1}^{d} n_j$ is the size of $\mathbf{X}^{\mathrm{FG}}$, $n_j$ is the size of $\mathbf{X}^{(j)}$. We can also note that entries of vector $\bigotimes_{j=1}^{d}\boldsymbol{\phi}_j(\cdot)$ are products of one-dimensional KPs. Therefore, similar to the one-dimensional case, $\bigotimes_{j=1}^{d}\boldsymbol{\phi}_j(\cdot)$ is a vector of compactly supported functions.

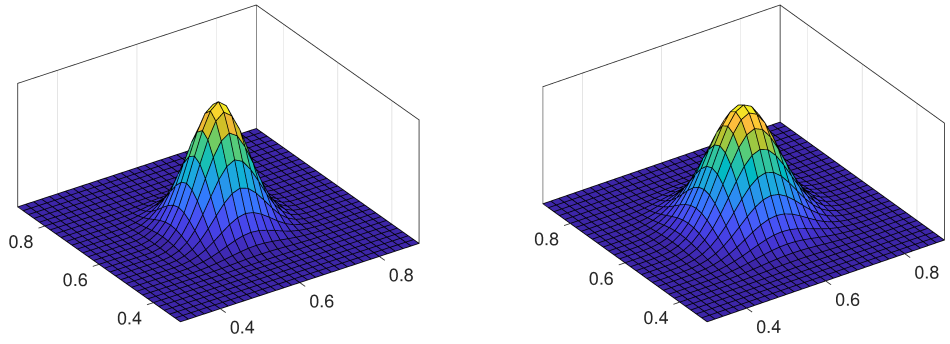


Figure 3: product KP basis functions $\phi_{(.4\ .5\ .6\ .7\ .8)}(x_1)\phi_{(.4\ .5\ .6\ .7\ .8)}(x_2)$ corresponding to Matérn-3/2 (left) and $\phi_{(.3\ .4\ .5\ .6\ .7\ .8\ .9)}(x_1)\phi_{((.3\ .4\ .5\ .6\ .7\ .8\ .9)}(x_2)$ corresponding to Matérn-5/2 (right) correlation function.

In (24)–(26), $\prod_{j=1}^{d}\boldsymbol{\phi}_j^T(x_j^*)\left[\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right]^{-1}K_j(\mathbf{X}^{(j)},x_j^*)$ and $\sum_{j=1}^{d}\frac{n}{n_j}(\log\det\boldsymbol{\phi}_j(\mathbf{X}^{(j)})-\log\det\mathbf{A}_j)$ can clearly be computed in $\mathcal{O}(\sum_{j=1}^{d}k^3n_j)$ time. The computation of $\left(\bigotimes_{j=1}^{d}\left[\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\right]^{-1}\right)\mathbf{v}$ for $\mathbf{v}\in\mathbb{R}^n$ is known as *Kronecker product least squares* (KLS), which has been extensively studied; see, e.g., Graham (2018); Fausett and Fulton (1994). Essentially, the task can be accomplished by sequentially solving $d$ problems where the $j^{\text{th}}$ problem is to solve $n/n_j$ independent banded linear equations with $n_j$ variables. With a banded solver, the total time complexity is $\mathcal{O}(dk^3n)$.

Although full grid designs result in simple and fast computation of GP regression, their sizes increase exponentially in the dimension. When the dimension is large, another class of grid-based designs called the *sparse grids* can be practically more useful. Let $\mathbf{X_l}$ denote full grids in

the form $\mathbf{X_l} = \times_{j=1}^{d} \mathbf{X}_{l_j}$ where $l_j \in \mathbb{N}$ and $\mathbf{X}_{l_j}$ is a one-dimensional point set satisfying the nested structure $\emptyset = \mathbf{X}_0 \subseteq \mathbf{X}_1 \subseteq \cdots \subseteq \mathbf{X}_{l_j}$ for each $j$. A sparse grid of level $\eta$ is defined as a union of full grids $\mathbf{X_l}$'s: $\mathbf{X}_\eta^{\mathrm{SG}} = \bigcup_{|\mathbf{l}|\leq \eta+d-1} \mathbf{X_l}$, where $|\mathbf{l}| := \sum_{j=1}^{d} l_j$. GP regression on sparse grid designs was first discussed in Plumlee (2014). According to Algorithm 1 in Plumlee (2014), (24) and (25), GP regression on $\mathbf{X}_\eta^{\mathrm{SG}}$ admits the expression

$$\mathbb{E}\left[Y(\mathbf{x}^*)\middle|\mathbf{Y}\right] = \mu(\mathbf{x}^*) + \sum_{|\mathbf{l}|=\max\{d,\eta-d+1\}}^{\eta} (-1)^{\eta-|\mathbf{l}|}\binom{d-1}{|\eta|-\mathbf{l}}\bar{f}_{\mathbf{l}}(\mathbf{x}^*), \tag{27}$$

$$\mathrm{Var}\left[Y(\mathbf{x}^*)\middle|\mathbf{Y}\right] = \sigma^2\Bigg(K(\mathbf{x}^*,\mathbf{x}^*) - \sum_{|\mathbf{l}|=\max\{d,\eta-d+1\}}^{\eta} (-1)^{\eta-|\mathbf{l}|}\binom{d-1}{|\eta|-\mathbf{l}}\bar{K}_{\mathbf{l}}(\mathbf{x}^*,\mathbf{x}^*)\Bigg), \tag{28}$$

where

$$\bar{f}_{\mathbf{l}} := \left(\bigotimes_{j=1}^{d} \boldsymbol{\phi}_{l_j}^T(x_j^*)\right)\left(\bigotimes_{j=1}^{d}\left[\boldsymbol{\phi}_{l_j}(\mathbf{X}_{l_j})\right]^{-1}\right)(\mathbf{Y_l} - \boldsymbol{\mu}_{\mathbf{l}}),$$

$$\bar{K}_{\mathbf{l}}(\mathbf{x}^*,\mathbf{x}^*) := \prod_{j=1}^{d} \boldsymbol{\phi}_{l_j}^T(x_j^*)\left[\boldsymbol{\phi}_{l_j}(\mathbf{X}_{l_j})\right]^{-1} K_j(\mathbf{X}_{l_j}, x_j^*),$$

come from (24) and (25), respectively; $\mathbf{Y_l}$ and $\boldsymbol{\mu}_{\mathbf{l}}$ denote the sub-vectors of $\mathbf{Y}$ and $\boldsymbol{\mu}$ on full grid $\mathbf{X_l}$, respectively. Based on Theorem 1 and Algorithm 2 in Plumlee (2014) and (26), $\log\det\mathbf{K}$ and $(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})^T\mathbf{K}^{-1}(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})^T$ in the log-likehood function (6) can be decomposed as the following linear combinations, respectively:

$$\sum_{|\mathbf{l}|=\max\{d,\eta+d-1\}}^{\eta}\sum_{j=1}^{d}\left(\log\frac{\det\boldsymbol{\phi}_{l_j}(\mathbf{X}_{l_j})}{\det\boldsymbol{\phi}_{l_j}(\mathbf{X}_{l_j-1})} - \log\frac{\det\mathbf{A}_{l_j}}{\det\mathbf{A}_{l_j-1}}\right)\prod_{w\neq j}(n_{l_w} - n_{l_w-1}), \tag{29}$$

$$\sum_{|\mathbf{l}|=\max\{d,\eta+d-1\}}^{\eta}\frac{(-1)^{\eta-|\mathbf{l}|}}{\sigma^2}\binom{d-1}{|\eta|-\mathbf{l}}(\mathbf{Y_l}-\mathbf{F_l}\boldsymbol{\beta})^T\left(\bigotimes_{j=1}^{d}\mathbf{A}_{l_j}\right)\left(\bigotimes_{j=1}^{d}\left[\boldsymbol{\phi}_{l_j}(\mathbf{X}_{l_j})\right]^{-1}\right)(\mathbf{Y_l}-\mathbf{F_l}\boldsymbol{\beta}), \tag{30}$$

where $\boldsymbol{\phi}_0(\mathbf{X}_0) = \mathbf{A}_0 = 1$ and $\mathbf{F_l}$ denotes the sub-matrix of $\mathbf{F}$ on full grid $\mathbf{X_l}$.

The above idea of direct computation fails to work for noisy data, because the Kronecker product structure of the covariance matrices breaks down due to the noise. Nonetheless, *conjugate gradient methods* can be implemented efficiently in the presence of the KP factorization (12). We defer the details to Section 5.

## 4. Numerical Experiments

We first conduct numerical experiments to assess the performance of the proposed algorithm for grid-based designs on test functions in Section 4.1. Next we employ the proposed method to one-dimensional real datasets in Section 4.2 to further assess its performance.

### 4.1 Grid-based Designs

In this section, we examine the performance of the proposed algorithm by synthetic functions over full and sparse grid designs.

#### 4.1.1 FULL GRID DESIGNS

We test our algorithm on the following deterministic function:

$$f(\mathbf{x}) = \sin(12\pi x_1) + \sin(12\pi x_2), \quad \mathbf{x} \in (0,1)^2.$$

Samples of $f$ are collected from a level-$\eta$ full grid design: $\mathbf{X}_\eta^{\mathsf{FG}} = \times_{j=1}^2 \{2^{-\eta}, 2\cdot 2^{-\eta}, \dots, 1-2^{-\eta}\}$ with $\eta = 5, 6, \cdots, 13$. The proposed KP algorithm is applied for GP regression using product Matérn correlation function. We choose the same correlation function in each dimension, with $\omega = 1$, and either $\nu = 3/2$ or $5/2$. We will investigate the *mean squared error* (MSE) and the average computational time over 1000 random test points for each prediction resulting from KP and the following approximation/fast GP regression algorithms with fixed correlation functions.

1. **laGP** R package [2]. In each experiment, laGP is run under Gaussian covariance family, the only covariance family supported by the package; size of the local subset is set as 100.

2. **Inducing Points** provided in the *GPML* tool box (Rasmussen and Nickisch, 2010). The number of inducing points $m$ is set as $m = \sqrt{n}$, which is the choice to achieve the optimal approximation power for Matérn-$5/2$ correlation according to Burt et al. (2019). However, if the algorithm crashes due to large sample size, $m$ is reduced to a level that the algorithm can run properly. We consider Matérn-$3/2$ and $5/2$ correlations.

3. **RFF** to approximate Matérn-$3/2$ and Matérn-$5/2$ correlation functions by feature functions $\left[\frac{1}{\sqrt{m}}(\cos\gamma_i x + b_i)\right]_{i=1}^m$, where $m = \sqrt{n}$, $\{\gamma_i\}_{i=1}^m$ are independent and identically distributed (i.i.d.) samples from $t$-distributions with degrees of freedom three and five, respectively, and $\{b_i\}_1^m$ are i.i.d. samples from the uniform distribution on $[0, 2\pi]$. If the algorithm crashes due to large sample size, $m$ is reduced to a level that the algorithm can run properly.

4. **Toeplitz** system solver incorporates the one-dimensional Toeplitz method and the Kronercker product technique. We consider Matérn-$3/2$ and $5/2$ correlations. In this experiment we use equally spaced design points, so that the Toeplitz method can work.

We sample 1000 i.i.d. test points uniformly from $(0,1)^2$ for each experimental trial. Figure 4 compares the MSE and the computational time of all algorithms, both under logarithmic scales, for sample sizes $2^{2j}$, $j = 5, 6, \dots 13$.

The performance curves of some algorithms in Figure 4 are incomplete, because these algorithms fail to work at a certain sample size due to a runtime error, or the prediction MSE ceases to improve. In this case, we stop the subsequent experimental trials with larger sample sizes for these algorithms. Specifically, for sample size larger than $2^{16}$, laGP breaks down due to runtime errors. The MSE of Toeplitz ceases to improve at sample size $2^{20}$. For sample size larger than $2^{20}$, the number of random features for RFF is fixed at $m = 2^{10}$ and the number of inducing points for inducing points method is fixed at $m = 2^{10}$ for subsequent trials. Otherwise, both the inducing points method and RFF break down because the approximated covariance matrices are nearly singular. Because of their fixed $m$'s, the performances of RFF and inducing points

2. `https://bobby.gramacy.com/r_packages/laGP/`

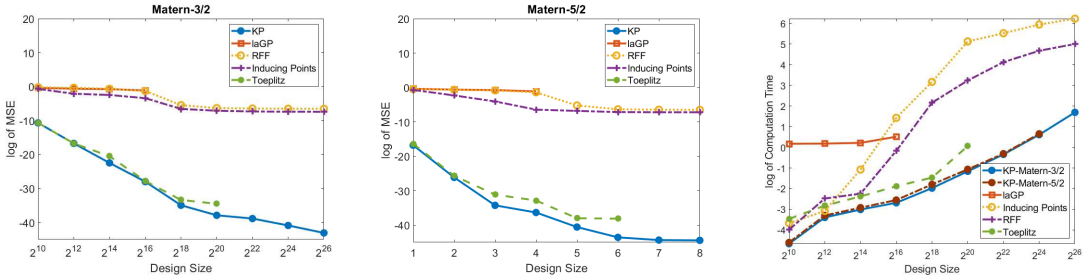


Figure 4: Logarithm of MSE for predictions with Matérn-3/2 correlation function (left) and Matérn-5/2 correlation function (middle) and logarithm of averaged computational time (right). The laGP uses the Gaussian covariance family in both the left and the middle figure. No results are shown for the cases when a runtime error occurs or the prediction error ceases to improve.

method do not have noticeable improvement for sample size larger than $2^{18}$. In contrast, KP can run on larger sample sets with sizes up to $2^{26}$ (more than 67 million) grid points.

It is shown in Figure 4 that KP has the lowest MSE and the fastest computational time in all experimental trials. The inducing points method, laGP and RFF have similar MSE in all experimental trials. The Toeplitz and KP algorithms, which compute the GP regression in exact ways, outperform other approximation methods.

### 4.1.2 SPARSE GRID DESIGNS

We test our algorithm on the *Griewank* function (Molga and Smutnicki, 2005), defined as

$$f(\mathbf{x}) = \sum_{j=1}^{d} \frac{x_j^2}{4000} - \prod_{j=1}^{d} \cos\left(\frac{x_j}{\sqrt{j}}\right) + 1, \quad \mathbf{x} \in (-2,2)^d,$$

with $d = 10$ and $d = 20$ respectively. Samples of $f$ are collected from a level-$\eta$ sparse grid design ($\eta = 3, 4, \cdots, 7$). We consider a constant mean $\mu(\mathbf{x}) = \beta$ and Matérn correlations with $\nu = 3/2, 5/2$ and a single scale parameter $\omega$ for all dimensions. We treat mean $\beta$, variance $\sigma^2$ and scale $\omega$ as unknown variables and use the *MLE-predictor*. We compare the performance of proposed KP algorithm and the direct method for GP regression on the sparse grids given in Plumlee (2014).

We sample 1000 i.i.d. points uniformly from the input space for each experimental trial. The mean squared error is estimated from these test points. In each trial, the mean squared errors of KP and direct method are in the same order and their differences are within $\pm 10^{-10}$. This is because both methods compute the MLE-predictor in an exact manner, and this also ensures the numerical correctness of the proposed method.

Figure 5 compares the logarithm of the needed computational time to estimate the unknown parameters $\beta$, $\sigma^2$ and $\omega$ and make predictions on the test points. The proposed KP algorithm is significantly advantageous in the computational time. When there are more than $10^5$ training points, the KP algorithm is at least twice faster than the direct method in each trial.

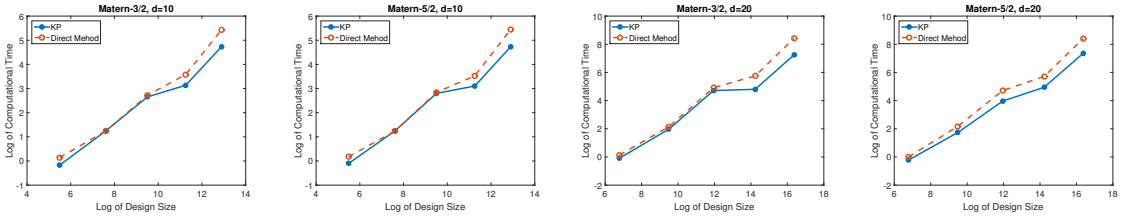


Figure 5: Logarithm of the computational time for MLE-predictions with Matérn-3/2 correlation function and Matérn-5/2 correlation with 10 and 20 dimensional inputs.

## 4.2 Real Datasets

In this section, we assess the performance of the proposed algorithm on two real-world datasets: the Mauna Loa $CO_2$ dataset (Keeling and Whorf, 2005) and the intraday stock prices of Apple Inc.

### 4.2.1 $CO_2$ DATA INTERPOLATION

This dataset consists of the monthly average atmosphere $CO_2$ concentrations at the the Mauna Loa Observatory in Hawaii for the last sixty years. The dataset has in total 767 data points and features a overall upward trend and a yearly cycle.

We fit the data using GP models reinforced by KP, inducing points, and RFF methods, respectively. For the proposed KP method, we consider a constant mean $\mu(\mathbf{x}) = \beta$, a single scale parameter $\omega$, and Matérn correlations with $\nu = 3/2, 5/2$, respectively. For the inducing points method and RFF, we consider also constant mean $\mu(\mathbf{x}) = \beta$. Different from KP, we use Gaussian correlations with scale parameter $\omega$ for the inducing points method and RFF. The number of inducing points is set as 100 for inducing points method. The number of generated random feature is set as 30 for RFF. We treat mean $\beta$, variance $\sigma^2$ and scale $\omega$ for all algorithms as unknown variables and use the *MLE-predictor*. For each algorithm, we compute the conditional mean and standard deviation on 2000 test points and plot the predictive curve. To evaluate the speed in training and prediction, we record the elapsed times for training and calculate the average time for a new prediction.

The training and prediction time of each algorithm is shown in Table 2. It is seen that the KP methods with Matérn-3/2 and Matérn-5/2 are faster than the inducing points and RFF. The predictive curve given by each algorithm is shown in Figure 6. Clearly, both KP methods interpolate adequately from 1960 to 2020 with accurate conditional standard deviations. In contrast, inducing points and RFF fail to interpolate the data, because the numbers of feature functions in inducing points and RFF are less than the number of observations. This results in predictive curves with higher standard deviations.

### 4.2.2 STOCK PRICE REGRESSION

This dataset consists of the intraday stock prices of Apple Inc from January, 2009 to April, 2011. The dataset has in total 1259 data points. We assume that the data points are corrupted by noise

| | $CO_2$ | | Stock Price | |
|---|---|---|---|---|
| Algorithm | $T_{\text{train}}$ (sec) | $T_{\text{pred}}$ ($10^{-3}$ sec) | $T_{\text{train}}$ (sec) | $T_{\text{pred}}$ ($10^{-3}$ sec) |
| KP Matérn-3/2 | $0.18 \pm 0.13$ | $2.84 \pm 0.96$ | $0.37 \pm 0.11$ | $2.83 \pm 1.07$ |
| KP Matérn-5/2 | $0.23 \pm 0.17$ | $3.31 \pm 1.22$ | $0.44 \pm 0.27$ | $3.54 \pm 1.73$ |
| Inducing Points | $0.28 \pm 0.09$ | $5.26 \pm 1.56$ | $0.58 \pm 0.13$ | $9.88 \pm 3.37$ |
| RFF | $0.25 \pm 0.12$ | $3.34 \pm 1.39$ | $0.50 \pm 0.26$ | $7.42 \pm 0.99$ |

Table 2: Comparisons of training and prediction time

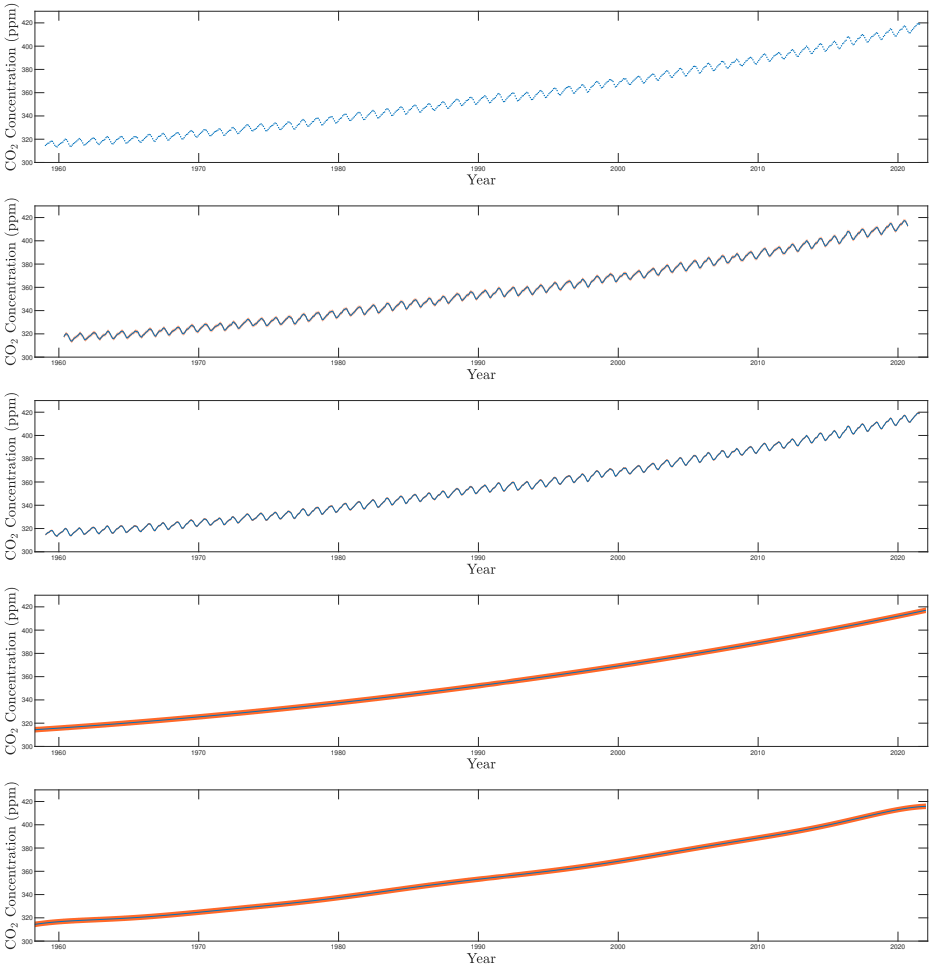


Figure 6: Observations of monthly $CO_2$ concentration (first row) and its interpolation by KP with Matérn-3/2 (second row) , KP with Matérn-5/2 (third row), Inducing Points (fourth row), and RFF (fifth row). The blue curves label predictions and the red areas label $\pm 1$ standard deviation.

so they are randomly distributed around some underlying trend. In this experiment, our goal is to reconstruct the underlying trend via GP regression.

Similar to Section 4.2.1, we run KP with Matérn-3/2 and Matérn-5/2 correlations on the dataset and use inducing pFoints and RFF as our benchmark algorithms. Settings of all algorithms are exactly the same as Section 4.2.1 except that the number of inducing points is set as 200 and the number of generated random features is set as 100 for RFF. We further treat the data variance parameter $\sigma_Y^2$ as an unknown parameter and use the MLE predictor. We also record the elapsed times for training and calculate the average time for a new prediction.

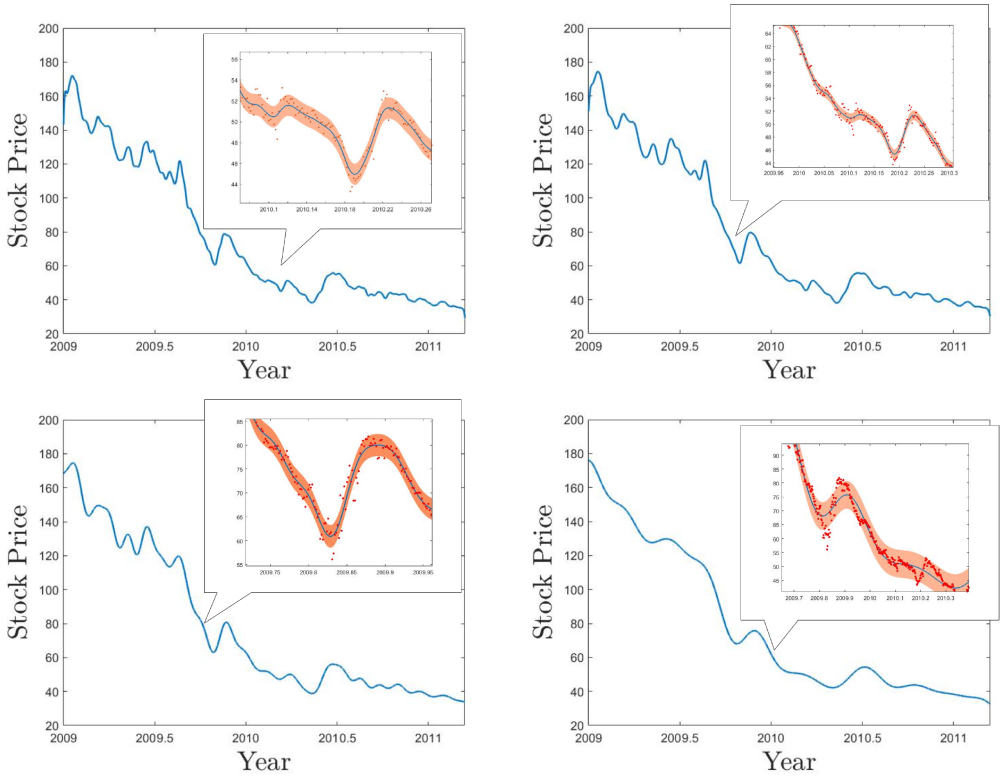


Figure 7: Stock Price regression by KP Matérn -3/2 (upper left), KP Matérn -5/2 (upper right), inducing points (lower left) and RFF (lower right). The blue curves label predictions, the red areas label $\pm 1$ standard deviations, and the red dots labels observations.

Similar to the previous experiment, Table 2 shows that the KP methods are more efficient than inducing points and RFF in both training and prediction. The predictive curve given by each algorithm is shown in Figure 7. We can see that both KP methods successfully capture the local changes of the overall trend while inducing points and RFF fail to do so. This is because neither inducing points nor RFF have enough number of feature functions to reconstruct curves with highly local fluctuations, and therefore, their predictive curves are too smooth so that a larger number of data points are distributed outside of their $\pm 1$ standard deviation areas.

## 5. Possible Extensions

Although the primary focus of this article is on exact algorithms, we would like to mention the potential of combining the proposed method with existing approximate algorithms. In this section, we will briefly discuss how to use the conjugate gradient method in the presence of the KP factorization (12) to accommodate a broader class of multi-dimensional Gaussian process regression problems.

### 5.1 Multi-dimensional GP Regression with Noisy Data

Suppose the input points lie in a full grid $\mathbf{X}^{\text{FG}}$ and the observed data $\mathbf{Z}$ is noisy: $Z(\mathbf{x}_i) = Y(\mathbf{x}_i)+\varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma_Y^2)$. Following arguments similar to what we have done in Sections 3.2 and 3.3, we can show that the following matrix operations are essential in computing the posterior and MLE:

$$\left[\bigotimes_{j=1}^{d} \boldsymbol{\phi}_j(\mathbf{X}^{(j)}) + \frac{\sigma_Y^2}{\sigma^2}\bigotimes_{j=1}^{d} \mathbf{A}_j\right]^{-1} \mathbf{v} \tag{31}$$

$$\log\det\left(\bigotimes_{j=1}^{d} \boldsymbol{\phi}_j(\mathbf{X}^{(j)}) + \frac{\sigma_Y^2}{\sigma^2}\bigotimes_{j=1}^{d} \mathbf{A}_j\right) \tag{32}$$

for some $\mathbf{v} \in \mathbb{R}^n$. The direct Kronecker product approach fails to work in this scenario because the additive noise breaks the tensor product structure. Nonetheless, conjugate gradient methods such as those implemented in GPyTorch (Gardner et al., 2018) or MATLAB (Barrett et al., 1994) can be employed to solve (31) and (32) efficiently. This is because the conjugate gradient methods require nothing more than the multiplication between the covariance matrix and a vector. In our case, both $\{\boldsymbol{\phi}_j(\mathbf{X}^{(j)})\}$ and $\{\mathbf{A}_j\}$ are banded matrices so $\bigotimes_{j=1}^{d} \boldsymbol{\phi}_j(\mathbf{X}^{(j)})$ and $\bigotimes_{j=1}^{d} \mathbf{A}_j$, which are Kronecker products of banded matrices, have only $\mathcal{O}(n)$ non-zero entries. Therefore, the cost of matrix-multiplications by the matres $\bigotimes_{j=1}^{d} \boldsymbol{\phi}_j(\mathbf{X}^{(j)})$ and $\bigotimes_{j=1}^{d} \mathbf{A}_j$ both scale *linearly* with respect to the number of points in the grid. If input points lie in a sparse grid, the posterior and MLE conditional on noisy data can also be computed efficiently because they can be decomposed as linear combinations of posteriors and MLEs on full grids as shown in section 3.3.

### 5.2 Additive Covariance Functions

Suppose the GP is equipped with the following additive covariance function:

$$K(\mathbf{x}, \mathbf{x}') = \sum_{\mathcal{J}\in\mathcal{F}} \prod_{j\in\mathcal{J}} k_{\mathcal{J},j}(x_j, x_j') \tag{33}$$

where $\mathcal{F}$ is any subset of the power set of $\{1, 2, \cdots, d\}$ and $k_{\mathcal{J},j}$ is any one-dimensional Matérn correlation with half-integer smoothness. When input points lie in a full grid $\mathbf{X}^{\text{FG}}$, the posterior and MLE can also be efficiently computed using conjugate gradient methods. In this case, the

covariance matrix can be written as the following form:

$$\mathbf{K} = \sum_{\mathcal{J}\in\mathcal{F}} \left[ \bigotimes_{j\in\mathcal{J}} \boldsymbol{\phi}_{\mathcal{J},j}(\mathbf{X}^{(j)}) \right] \left[ \bigotimes_{j\in\mathcal{J}} \mathbf{A}_{\mathcal{J},j}^{-1} \right]. \tag{34}$$

The matrix-multiplication $\mathbf{Kv}$ can be computed in linear time in the size of $\mathbf{X}^{\mathrm{FG}}$ for any vector $\mathbf{v} \in \mathbb{R}^n$. Firstly, we use KLS techniques introduced in Section 3.3 to compute $\mathbf{v}' = \big[ \bigotimes_{j\in\mathcal{J}} \mathbf{A}_{\mathcal{J},j}^{-1} \big]\mathbf{v}$, which has linear time complexity in the size of $\mathbf{X}^{\mathrm{FG}}$. Then, we can compute $\big[ \bigotimes_{j\in\mathcal{J}} \boldsymbol{\phi}_{\mathcal{J},j}(\mathbf{X}^{(j)}) \big]\mathbf{v}'$, which has the same time complexity. Similar to Section 5.1, efficient algorithms also exist when input points lie in a sparse grid.

## 6. Conclusions and Discussion

In this work, we propose a rapid and exact algorithm for one-dimensional Gaussian process regression under Matérn correlations with half-integer smoothness. The proposed method can be applied to some multi-dimensional problems by using tensor product techniques, including grid and sparse grid designs, and their generalizations (Plumlee et al., 2021). With a simple modification, the proposed algorithm can also accommodate noisy data. If the design is not grid-based, the proposed algorithm is not applicable. We may apply the idea of Ding et al. (2020) to develop approximated algorithms, which work for not only regression problems, but also for other type of supervised learning tasks.

Another direction for future work is to establish the relationship between KP and the *state-space approaches*. The latter methods leverage the Gauss-Markov process representation of certain GPs, including Matérn-type GPs with half-integer smoothness, and employ the Kalman filtering and related methodologies to handle GP regression, which results in a learning algorithm with time and space complexity both in $O(n)$ (Hartikainen and Särkkä, 2010; Saatçi, 2012; Sarkka et al., 2013; Loper et al., 2021). Whether the key mathematical theories of KP and state-space approaches are essentially equivalent is unknown and requires further investigation. Although having the same time and space complexity as KP, the Kalman filtering method is formulated in a sequential data processing form, which significantly differs from the usual supervised learning framework and makes it more difficult to comprehend. The proposed method, in contrast, is presented by a simple matrix factorization (12), which is easy to implement and incorporated in more complicated models.

## Acknowledgements

The authors are grateful to two referees and the Associate Editor for very helpful comments and suggestions. This research is supported by NSF DMS-1914636 and CCF-1934904.

# Appendix

## Appendix A. Paley-Wiener Theorems

We will need two Paley-Wiener theorems in our proofs, given by Lemmas 15 and 16. For detailed discussion, we refer to Chapter 4 of Stein and Shakarchi (2003). Denote the support of function $f$ as $\operatorname{supp} f$.

**Definition 14 (Stein and Shakarchi (2003), page 112)** *We say that a function $f$ is of **moderate decrease** if there exists $M \in \mathbb{R}$ so that $|f(x)| \leq M/(1+|x|^{\alpha})$ for some $\alpha > 1$, for all $x \in \mathbb{R}$.*

**Lemma 15 (Theorem 3.3 in Chapter 4 of Stein and Shakarchi (2003))** *Suppose $f$ is continuous and of moderate decrease on $\mathbb{R}$, $\hat{f}$ is the Fourier transform of $f$. Then, $f$ has an extension to the complex plane that is entire with*[3] *$|f(z)| \leq Ae^{M|z|}$ for some $A > 0$, if and only if* $\operatorname{supp} \hat{f} \subset [-M, M]$.

**Lemma 16 (Theorem 3.5 in Chapter 4 of Stein and Shakarchi (2003))** *Suppose $f$ is continuous and of moderate decrease on $\mathbb{R}$, $\hat{f}$ is the Fourier transform of $f$. Then* $\operatorname{supp} \hat{f} \subset [0, +\infty)$ *if and only if $f$ can be extended to a continuous and bounded function in the closed upper half-plane $\{z = x + iy : y \geq 0\}$ with $f$ holomorphic in the interior.*

## Appendix B. Technical Proofs

### B.1 Algebraic Properties

The following Lemma 17 will be useful in proving the main theorems. We use $\deg p$ to denote the degree of polynomial $p$. For notational convenience, we define the degree of the zero polynomial as $-1$. We say $x$ a *zero* of function $f$ if $f(x) = 0$.

**Lemma 17** *Let $p_1$ and $p_2$ be polynomials with* $\deg p_1 = d_1, \deg p_2 = d_2$. *If* $\max(\deg p_1, \deg p_2) \geq 0$ *and $c \neq 0$, then the function $f(x) := p_1(x)e^{cx} + p_2(x)e^{-cx}$ has at most $d_1 + d_2 + 1$ real-valued zeros.*

**Proof** Without loss of generality, we assume that $p_1$ is non-zero. Suppose $f$ has at least $d_1 + d_2 + 2$ real-valued zeros. Equivalently, the function $g(x) = p_1(x)e^{2cx} + p_2(x)$ has at least $d_1 + d_2 + 2$ real-valued zeros. The mean value theorem implies $g'(\xi) = 0$ for some $\xi$ lying between two consecutive real-valued zeros of $g$. Therefore, $g'$ has at least $d_1 + d_2 + 1$ real-valued zeros. Repeating this procedure $d_2 + 1$ times, we can conclude that $g^{(d_2+1)}$ has at least $d_1 + 1$ real-valued zeros. Note that $g^{(d_2+1)}$ possesses the form $g^{(d_2+1)}(x) = q(x)e^{2cx}$, where $q(x)$ is a non-zero polynomial with degree $d_1$. Because $e^{2cx} > 0$, $q(x)$ has at least $d_1 + 1$ real-valued zeros, which contradicts the fundamental theorem of algebra. ■

Lemma 18 can directly lead to Theorems 2 and 9. We call the vector of zero the *trivial solution* to a homogeneous system of linear equations.

**Lemma 18** *The following homogeneous systems have only the trivial solutions.*

1. *For $m \geq 1$, the $m \times m$ system about $(u_1, \ldots, u_m)^T$:*

$$\sum_{j=1}^{m} b_j^l \exp\{cb_j\} u_j = 0,$$

*with $l = 0, \ldots, m-1$, $c \neq 0$ and distinct real numbers $b_1, \ldots, b_m$.*

3. Stein and Shakarchi (2003) uses an equivalent but different definition of the inverse Fourier transform as $\tilde{f}(\xi) = \int_{-\infty}^{\infty} f(x)e^{2\pi i\xi} dx$, so that this inequality becomes $|f(z)| \leq Ae^{2\pi M|z|}$.

2. *For $m, s \geq 1$, the $(m+s)\times(m+s)$ system about $(u_1, \ldots, u_{m+s})^T$:*

$$\sum_{j=1}^{m+s} b_j^l \exp\{cb_j\}u_j = 0, \quad \sum_{j=1}^{m+s} b_j^r \exp\{-cb_j\}u_j = 0, \tag{35}$$

*with $l = 0, \ldots, m-1$, $r = 0, \ldots, s-1$, $c \neq 0$ and distinct real numbers $b_1, \ldots, b_{m+s}$.*

**Proof** For both parts, it suffices to prove that the coefficient matrices are of full row ranks, which is equivalent to that they are of full column ranks. This inspires us to consider the transpose of the coefficient matrices.

Here we only provide the proof for Part 2. The proof for Part 1 follows from similar lines. For Part 2, the linear system corresponding to the transposed coefficient matrix is

$$\sum_{l=0}^{m-1} b_j^l \exp\{cb_j\}v_l + \sum_{r=0}^{s-1} b_j^r \exp\{-cb_j\}v_{m+r} = 0, \tag{36}$$

with the vector of unknowns $(v_0, \ldots, v_{m-1})^T$. Suppose (35) has a non-trivial solution. Then (36) also has a nontrivial solution, denoted as $(v_0^*, \ldots, v_{m+s-1}^*)^T$. Write $p_1(x) = \sum_{l=0}^{m-1} v_l^* x^l$ and $p_2(x) = \sum_{r=0}^{s-1} v_{m+r}^* x^r$. Therefore, (36) implies that each $b_j$ is a zero of the function $f(x) := p_1(x)e^{cx} + p_2(x)e^{-cx}$. Hence $f(x)$ has at least $m+s$ distinct zeros. Note that $\deg p_1 \leq m-1, \deg p_2 \leq s_1$. Because $(v_0^*, \ldots, v_{m+s-1}^*)^T$ is non-trivial, we have $\max(\deg p_1, \deg p_2) \geq 0$. Thus Lemma 17 yields that $f(x)$ has no more than $m+s-1$ distinct zeros, a contradiction. ■

**Proof** [Proof of Theorem 2] This theorem follows directly from Part 2 of Lemma 18, because each $(k-1)\times(k-1)$ submatrix of the coefficient matrix corresponds a linear system of the form in Part 2 of Lemma 18. ■

**Proof** [Proof of Theorem 9] This theorem follows directly from Lemma 18, because each $(s-1)\times(s-1)$ submatrix of the coefficient matrix corresponds a linear system of the form in of Lemma 18. ■

**Proof** [Proof of Theorem 3] Let $(A_1, \ldots, A_k)^T$ be a solution to (8). It suffices to prove that

$$\sum_{j=1}^{k} A_j(a_j + t)^l \exp\{\delta c(a_j + t)\} = 0,$$

for $l = 0, \ldots, (k-3)/2$, $\delta = \pm 1$ and each $t \in \mathbb{R}$. This can be proved by noting that

$$\begin{aligned} & \sum_{j=1}^{k} A_j(a_j + t)^l \exp\{\delta c(a_j + t)\} \\ = \quad & \exp\{\delta ct\} \sum_{j=1}^{k} A_j \exp\{\delta ca_j\} \sum_{m=0}^{l} \binom{l}{m} a_j^m t^{l-m} \end{aligned}$$

$$= \exp\{\delta c t\} \sum_{m=0}^{l} \binom{l}{m} t^{l-m} \left( \sum_{j=1}^{k} A_j a_j^m \exp\{\delta c a_j\} \right) = 0,$$

where the last equality follows from the identity $\sum_{j=1}^{k} A_j a_j^m \exp\{\delta c a_j\} = 0$ for $0 \le m \le l$, ensured by equation system (8). ■

**Proof** [Proof of Theorem 10] The proof follows from arguments similar to the proof of Theorem 3. ■

### B.2 Results for the Supports

We first prove the following useful lemma.

**Lemma 19** *Let $K$ be a Matérn correlation with a half-integer smoothness. Suppose $b_1 < \cdots < b_m$ and $t \in (b_\tau, b_{\tau+1})$ for some $1 \le \tau < n$. Let $\psi(x) = \sum_{j=1}^{m} B_j K(x, b_j)$, for $B_j \in \mathbb{R}$. Denote $\psi_1(x) = \sum_{j=1}^{\tau} B_j K(x, b_j)$, and $\psi_2(x) = \sum_{j=\tau+1}^{m} B_j K(x, b_j)$. If there exists $\epsilon > 0$ such that for $x \in (t-\epsilon, t+\epsilon)$, $\psi(x) = 0$, then*

$$\psi_1(x) = \begin{cases} \psi(x), & \text{for } x < b_\tau, \\ 0, & \text{otherwise.} \end{cases} \quad \text{and} \quad \psi_2(x) = \begin{cases} 0, & \text{for } x \le b_{\tau+1}, \\ \psi(x), & \text{otherwise.} \end{cases}$$

**Proof** It is known that when $\nu = p + 1/2$ with $p \in \mathbb{N}$, the Matérn correlation can be expressed as (Santner et al., 2003)

$$K(x, x') = P_p(|x - x'|) \exp\{-c|x - x'|\}, \tag{37}$$

where $c = \sqrt{2\nu/\omega}$ and $P_p(x) = \sigma^2 \frac{p!}{(2p)!} \sum_{j=0}^{p} \frac{(p+j)!}{j!(p-j)!} (2cx)^{p-j}$ is a polynomial of degree $p$.

Therefore, for any $x \in (b_\tau, b_{\tau+1})$,

$$\begin{aligned} \psi(x) &= \psi_1(x) + \psi_2(x) \\ &= \sum_{j=1}^{\tau} B_j P_p(x - b_j) e^{-c(x-b_j)} + \sum_{j=\tau+1}^{m} B_j P_p(b_j - x) e^{c(x-b_j)} \\ &=: p_1(x) e^{-cx} + p_2(x) e^{cx}, \end{aligned}$$

where $p_1$ and $p_2$ are polynomials. Thus $\psi(x)$ is an analytic function on $(b_\tau, b_{\tau+1})$. Then $\psi(x) = 0$ for $x \in (t-\epsilon, t+\epsilon)$ implies $\psi(x) = 0$ for $x \in (b_\tau, b_{\tau+1})$, which is possible only if $p_1 \equiv p_2 \equiv 0$, because otherwise $\psi$ can only have at most $2p+1$ distinct zeros on $(b_\tau, b_{\tau+1})$ according to Lemma 17. Hence, $\psi_1(x) = 0$ whenever $x \ge b_\tau$, and $\psi_2(x) = 0$ whenever $x \le b_{\tau+1}$. ■

The following lemma formalizes the rationale behind (8).

**Lemma 20** *Let $U$ be a connected open subset of $\mathbb{C}$ containing a point $z_0$. Let $f(z)$ be a holomorphic function on $U$, and $m$ a positive integer. Then $f(z)(z - z_0)^{-m}$ can be extended as a holomorphic function on $U$ if and only if $f^{(j)}(z_0) = 0$, for $j = 0, \ldots, m-1$,*

**Proof** First assume $f(z)(z-z_0)^{-m}$ being holomorphic. Then $f(z_0)$ must be zero, because otherwise $\lim_{z\to z_0} f(z)(z-z_0)^{-m} = \infty$. If $f$ vanishes identically in $U$, the desired result is trivial. If $f$ does not vanish identically in $U$, according to Theorem 1.1 in Chapter 3 of Stein and Shakarchi (2003), there exists a neighborhood $V \subset U$ of $z_0$, and a unique positive integer $m'$ such that $f(z) = (z-z_0)^{m'} g(z)$ for $z \in V$ with $g$ being a non-vanishing holomorphic function on $V$. Clearly it must hold that $m' \geq m$, because otherwise we have $\lim_{z\to z_0} f(z)(z-z_0)^{-m} = \infty$ again. Then it is easily checked that $f^{(j)}(z_0) = 0$, for $j = 0, \ldots, m-1$.

For the converse, in a small disc centered at $z_0$ the function $f$ has a power series expansion $f = \sum_{j=0}^{\infty} a_j (z-z_0)^j$, where $a_j = f^{(j)}(z_0)/j!$ for each $j \in \mathbb{N}$. Thus $a_0 = \cdots = a_{m-1} = 0$. Consequently, $f(z)(z-z_0)^{-m} = \sum_{j=m}^{\infty} a_j (z-z_0)^{j-m}$ and thus is holomophic on $U$. ■

**Proof** [Proof of Theorem 8] As before, let $k := 2\nu + 2$. Suppose that $\phi(x) = \sum_{j=1}^{m} A_j K(x, a_j)$ has a compact support. The analytic continuation of its inverse Fourier transform is

$$\tilde{\phi}(z) = \sum_{j=1}^{m} A_j \exp\{a_j z\}(c^2 + z^2)^{(k-1)/2} := \gamma(z)(c^2 + z^2)^{(k-1)/2},$$

for $z \in \mathbb{C} \in \{\pm ci\}$. Then Lemma 20 entails $\gamma^{(j)}(\pm ci) = 0$ for $j = 0, \ldots, (k-3)/2$, which leads to the linear system

$$\sum_{j=1}^{m} A_j a_j^l \exp\{\delta c a_j\} = 0,$$

with $l = 0, \ldots, (k-3)/2$ and $\delta = \pm 1$. But this system has only the trivial solution in view of Lemma 18. ■

**Proof** [Proof of Theorem 5]Without loss of generality, we can assume that $a_1 = -M$ and $a_k = M$ for some positive real number $M$, because otherwise we can apply a shift translation to convert the original problem to this form in view of Theorem 3.

We first employ Lemma 15 to show that $\operatorname{supp} \phi_{\mathbf{a}} \subset [-M, M] = [a_1, a_k]$. Lemma 20 implies that $\tilde{\phi}_{\mathbf{a}}$ is entire. By its continuity, $|\tilde{\phi}_{\mathbf{a}}|$ is bounded in the region $|z| \leq 2c$. For $|z| \geq 2c$, we have

$$\begin{aligned} \left|\tilde{\phi}_{\mathbf{a}}(z)\right| &= \left|\gamma(z)\right| \cdot \left|(c^2 + z^2)^{(k-1)/2}\right| \leq c^{k-1} \left|\gamma(z)\right| \\ &\leq c^{k-1} \sum_{j=1}^{k} \left|A_j\right| \cdot \left|\exp\{-i a_j z\}\right| \leq c^{k-1} \sum_{j=1}^{k} \left|A_j\right| \exp\{M|z|\}, \end{aligned}$$

where the last inequality follows from the fact that $|e^z| \leq e^{|z|}$. Clearly, $\tilde{\phi}_{\mathbf{a}}$ is of moderate decrease. According to Lemma 15, we obtain that $\operatorname{supp} \phi_{\mathbf{a}} \subset [-M, M]$.

It remains to prove that $\operatorname{supp} \phi_{\mathbf{a}} = [-M, M]$. Suppose $\operatorname{supp} \phi_{\mathbf{a}} \neq [-M, M]$. Then we can find $-M \leq M_1 < M_2 \leq M$, such that $\phi_{\mathbf{a}}(x) = 0$ for $x \in [M_1, M_2]$. Therefore, Lemma 19 implies that $\phi_{\mathbf{a}}$ can be expressed as

$$\phi_{\mathbf{a}}(x) = \sum_{j=1}^{\tau} A_j K(x, a_j) + \sum_{j=\tau+1}^{k} A_j K(x, a_j) := \psi_1(x) + \psi_2(x),$$

for some $1 \le \tau < n$, such that $\operatorname{supp}\psi_1 \subset [a_1, a_\tau], \operatorname{supp}\psi_2 \subset [a_{\tau+1}, a_n]$. Because either $\psi_1$ or $\psi_2$ must not be identically vanishing, such a function, according to Definition 1, is a KP with degree less than $k$. But this contradicts Theorem 8. ■

**Proof** [Proof of Theorem 7] For Part 1, when the smoothness parameter $\nu$ of a Matérn kernel is not a half integer, then direct calculations shows

$$\begin{aligned}\tilde{\phi}_a(x) &\propto \left[\sum_{j=1}^{k} A_j \exp\{ia_j x\}\right](c^2 + x^2)^{-\nu-\frac{1}{2}} \\ &= \left[\sum_{j=1}^{k} A_j \exp\{ia_j x\}\right]\exp\left\{-(\nu+\frac{1}{2})\log(x^2+c^2)\right\}. \end{aligned} \tag{38}$$

The goal is to prove that (38) cannot be extended to an entire function unless $A_j = 0$ for each $j$. There is no continuous complex logarithm function defined on all $\mathbb{C} \setminus \{0\}$. Here we consider the principal branch of the complex logarithm $\operatorname{Log} z := \log|z| + i \operatorname{Arg} z$, where $\operatorname{Arg} z$ is the principal value of the argument of $z$ ranging in $(-\pi, \pi]$. For $x \in \mathbb{R}$, we have $\log x = \operatorname{Log} x$. It is known that $\operatorname{Log} x$ is holomorphic on the set $\mathbb{C} \setminus \{z \in \mathbb{R} : z \le 0\}$. Therefore, the function in (38) can be analytically continued to the region $\mathcal{S} := \mathbb{C} \setminus \mathcal{S}^c := \mathbb{C} \setminus \{yi : y \in \mathbb{R}, |y| \ge c\}$.

Because analytical continuation is unique, the analytical continuation of (38) should coincide with

$$g(z) := \left[\sum_{j=1}^{k} A_j \exp\{ia_j z\}\right]\exp\left\{-(\nu+\frac{1}{2})\operatorname{Log}(z^2+c^2)\right\}$$

on $\mathcal{S}$. Because $\nu + 1/2$ is not an integer, $\exp\left\{-(\nu+\frac{1}{2})\operatorname{Log}(x^2+c^2)\right\}$ is discontinuous when $z$ moves across $\mathcal{S}^c$. Suppose there exists a KP. Then $g(z)$ must an entire function in view of lemma 15. To make $g(z)$ continuous on $\mathcal{S}^c$, we must have $\sum_{j=1}^{k} A_j \exp\{ia_j z\} = 0$ on $\mathcal{S}^c$, but this readily implies $\sum_{j=1}^{k} A_j \exp\{ia_j z\} = 0$ on $\mathbb{C}$ as $\sum_{j=1}^{k} A_j \exp\{ia_j z\}$ is also an entire function. Therefore $g(z) = 0$. By the uniqueness of the Fourier transform, the underlying KP vanishes identically in $\mathbb{R}$, which leads to a contradiction.

For part 2, a Gaussian correlation function $K$ is an analytic function on $\mathbb{R}$, and so does $\phi_{\mathbf{a}} := \sum_{j=1}^{n} A_j K(x - a_i)$. Therefore, $\phi_{\mathbf{a}}$ cannot have a compact support unless $\phi_{\mathbf{a}} \equiv 0$. ■

**Proof** [Proof of Theorem 11] Without loss of generality, we can assume that $a_1 = 0$ because otherwise we can apply shift translation to make this happen in view of Theorem 10.

Clearly, $\phi_{\mathbf{a}}$ is of moderate decrease in view of the expression (37). Direct calculation shows

$$\tilde{\phi}_{\mathbf{a}}(z) \propto \left[\sum_{j=1}^{s} A_j \exp\{ia_j z\}\right](c^2+z^2)^{-(k-1)/2} = \gamma(z)(c^2+z^2)^{-(k-1)/2}.$$

Equation (10) implies $\gamma^{(j)}(ci) = 0$ for $j = 0, \dots, (k-3)/2$. Thus $\frac{d^j}{dz^j}(\gamma(z)(z+ci)^{-(k-1)/2})\big|_{z=ci} = 0$ for $j = 0, \dots, (k-3)/2$, which, together with Lemma 20, yields that $f(z) := \gamma(z)(c^2+z^2)^{-(k-1)/2}$ is holomorphic in a neighborhood of $ci$. So $f(z)$ is continuous on the upper half-plane $\{z =$

$x+iy : y \geq 0\}$ and is holomorphic in its interior. To employ Lemma 16, it remains to proof that $f(z)$ is bounded in $\{z = x+iy : y \geq 0\}$. For $|z-ci| \leq c$, $f(z)$ is clearly bounded as it is a continuous function. For $|z-ci| \geq c$ and $z \in \{z = x+iy : y \geq 0\}$, we have

$$|(c^2+z^2)^{-(k-1)/2}| = |z-ci|^{-(k-1)/2}|z+ci|^{-(k-1)/2} \leq c^{-(k-1)}.$$

Write $z = x+iy$, then

$$|f(z)| = |\gamma(z)||(c^2+z^2)^{-(k-1)/2}| \leq \left|\sum_{j=1}^{s} A_j \exp\{ia_j(x+iy)\}c^{-(k-1)}\right|$$

$$\leq \quad c^{-(k-1)} \sum_{j=1}^{s} \left|A_j\right| \left|\exp\{ia_j(x+iy)\}\right| = c^{-(k-1)} \sum_{j=1}^{s} \left|A_j\right| \exp\{-a_j y\},$$

which is bounded as $y \geq 0$. Therefore, according to Lemma 16, $\operatorname{supp} \phi_{\mathbf{a}} \subset [0, +\infty)$.

Next, we prove that $0 \in \operatorname{supp} \phi_{\mathbf{a}}$. First, it can be shown that $A_1 \neq 0$, because otherwise $(A_2, \ldots, A_s)^T$ is a solution to the linear system $\sum_{j=2}^{s} a_j^l \exp\{-ca_j\}A_j = 0$, with $l = 0, \ldots, (k-3)/2$, if $s = (k+1)/2$, or $\sum_{j=2}^{s} a_j^l \exp\{-ca_j\}A_j = 0, \quad \sum_{j=2}^{s} a_j^r \exp\{ca_j\}A_j = 0$, with $l = 0, \ldots, (k-3)/2$ and $r = 0, \ldots, s-(k+3)/2$, if $s \geq (k+3)/2$. Then Lemma 17 suggests that $A_j = 0$ for all $j = 0, \ldots, s$, which is a contradiction. Now, suppose $0 \notin \operatorname{supp} \phi_{\mathbf{a}}$. Then there exists $\epsilon > 0$, such that $\phi_{\mathbf{a}}(x) = 0$ for all $x < \epsilon$. Without loss of generality, assume $\epsilon < a_2$. We now apply a shift transformation. Recall that $T_{-\epsilon}(\mathbf{a}) := (a_1 - \epsilon, \ldots, a_s - \epsilon)$. Theorem 10 implies that $\phi_{T_\epsilon(\mathbf{a})}(x) = \sum_{j=1}^{s} A_j K(x+\epsilon, a_j)$. Thus,

$$\tilde{\phi}_{T_\epsilon(\mathbf{a})}(z) \propto \left[\sum_{j=1}^{s} A_j \exp\{i(a_j-\epsilon)z\}\right](c^2+z^2)^{-(k-1)/2}.$$

It is easily seen that $\tilde{\phi}_{T_\epsilon(\mathbf{a})}(z)$ is unbounded if $z = iy$ for sufficiently large $y > 0$. Specifically,

$$\tilde{\phi}_{T_\epsilon(\mathbf{a})}(iy) \quad \propto \quad \left[\sum_{j=1}^{s} A_j \exp\{-(a_j-\epsilon)y\}\right](c^2-y^2)^{-(k-1)/2}$$

$$= \quad A_1(c^2-y^2)^{-\frac{k-1}{2}} \exp\{(\epsilon-a_1)y\} + (c^2-y^2)^{-\frac{k-1}{2}} \sum_{j=2}^{s} A_j \exp\{-(a_j-\epsilon)y\},$$

where the first term diverges and the second term converges to zero as $y \to +\infty$, because $A_1 \neq 0$ and $a_1 < \epsilon < a_2 < \cdots < a_s$.

The remainder is to prove that $\operatorname{supp} \phi_{\mathbf{a}} = [0, +\infty)$. Suppose $\operatorname{supp} \phi_{\mathbf{a}} \neq [0, +\infty)$. Then there exist $M_2 > M_1 > 0$, such that $\phi_{\mathbf{a}}(x) = 0$ whenever $x \in (M_1, M_2)$. Without loss of generality, we assume that $M_1, M_2 \notin \{a_1, \ldots, a_s\}$. Then we can write

$$\phi_{\mathbf{a}}(x) = \sum_{j=1}^{s} A_j K(x, a_j) + 0 \cdot K(x, M_1) + 0 \cdot K(x, M_2).$$

Then Lemma 19 implies that $\phi_{\mathbf{a}}(x)$ can be decomposed into

$$\phi_{\mathbf{a}}(x) = \sum_{j=1}^{\tau} A_j K(x, a_j) + \sum_{j=\tau+1}^{s} A_j K(x, a_j) := \psi_1(x) + \psi_2(x),$$

for some $1 \le \tau < s$, such that $\operatorname{supp}\psi_1 \subset [0, M_1]$ and $\operatorname{supp}\psi_1 \subset [M_2, +\infty)$. Because $0 \in \operatorname{supp}\phi_{\mathbf{a}}$, we have $\operatorname{supp}\psi_1 \neq \emptyset$. Therefore, $\psi_1$ is a non-zero function and has a compact support, which contradicts Theorem 8. ■

### B.3 Linear Independence

**Proof** [Proof of Theorem 13] We have learned from Theorems 5 and 11, and the analogous counterpart of Theorem 11 for the left-sided KPs that:

1. The left-sided KPs $\phi_1, \phi_2, \dots, \phi_{(k-1)/2}$ have supports $(-\infty, x_{(k+1)/2}]$, $(-\infty, x_{(k+1)/2+1}]$, ..., $(-\infty, x_{k-1}]$, respectively.
2. The KPs $\phi_{(k+1)/2}, \phi_{(k+1)/2+1}, \dots, \phi_{n-(k-1)/2}$ have supports $[x_1, x_k], [x_2, x_{k+1}]$, ..., $[x_{n-k+1}, x_n]$, respectively.
3. The right-sided KPs $\phi_{n-(k-3)/2}, \dots, \phi_{n-1}, \phi_n$ have supports $[x_{n-k+2}, \infty)$, ..., $[x_{n-(k-1)/2-1}, \infty)$, $[x_{n-(k-1)/2}, \infty)$ respectively.

Therefore, for $\tau < n - (k-1)/2$, any function of the form $f := \sum_{j=1}^{\tau} \lambda_j \phi_j$ satisfies $\operatorname{supp} f \subset (-\infty, x_{\tau+(k-1)/2}]$. Note that $\operatorname{supp}\phi_{\tau+1} = [x_{\tau+1-(k-1)/2}, x_{\tau+1+(k-1)/2}] \not\subset (-\infty, x_{\tau+(k-1)/2}]$, which proves that $\phi_{\tau+1} \notin \operatorname{span}\{\phi_1, \dots, \phi_\tau\}$. Hence, by induction, we can prove that $\phi_1, \dots, \phi_{n-(k-1)/2}$ are linearly independent.

Similarly, we can prove that the right-sided KPs $\phi_{n-(k-3)/2}, \dots, \phi_n$ are linearly independent. Now suppose $\sum_{j=1}^{n} \xi_j \phi_j = 0$ for $\xi_1, \dots, \xi_n \in \mathbb{R}$. We rearrange this identity as

$$f_1 := \sum_{j=1}^{n-(k-1)/2} \xi_j \phi_j = -\sum_{j=n-(k-3)/2}^{n} \xi_j \phi_j =: f_2, \tag{39}$$

i.e., the left-hand side of (39) is a linear combination of the left-sided KPs and the KPs, and the right-hand side of (39) is a linear combination of the right-sided KPs.

Note that $\operatorname{supp} f_1 \subset (-\infty, x_n]$ and $\operatorname{supp} f_2 \subset [x_{n-k+2}, +\infty)$. Then identify (39) implies $\operatorname{supp} f_2 \subset (-\infty, x_n] \cap [x_{n-k+2}, +\infty) = [x_{n-k+2}, x_n]$. By definition, $f_2$ is a linear combination of $k-1$ functions $K(\cdot, a_{n-k+2}), \dots, K(\cdot, a_n)$. Hence, by Theorem 8, $f_2$ has a compact support only if $f_2 \equiv 0$, which, together with the fact that $\phi_{n-(k-3)/2}, \dots, \phi_n$ are linearly independent, yields that $\xi_{n-(k-3)/2} = \cdots = \xi_n = 0$. Then by (39), we similarly have $\xi_1, \dots, x_{n-(k-1)/2} = 0$ because $\phi_1, \dots, \phi_{n-(k-1)/2}$ are proved to be linearly independent. In summary, we prove that $\phi_1, \dots, \phi_n$ are linearly independent. ■

## References

Kendall E Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, 2008.

Sudipto Banerjee, Bradley P Carlin, and Alan E Gelfand. *Hierarchical Modeling and Analysis for Spatial Data*. CRC press, 2014.

Richard Barrett, Michael Berry, Tony F Chan, James Demmel, June Donato, Jack Dongarra, Victor Eijkhout, Roldan Pozo, Charles Romine, and Henk Van der Vorst. *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.

Yakoub Bazi and Farid Melgani. Gaussian process approach to remote sensing image classification. *IEEE transactions on geoscience and remote sensing*, 48(1):186–197, 2009.

Thang D Bui, Josiah Yan, and Richard E Turner. A unifying framework for gaussian process pseudo-point approximations using power expectation propagation. *The Journal of Machine Learning Research*, 18(1):3649–3720, 2017.

David Burt, Carl Edward Rasmussen, and Mark Van Der Wilk. Rates of convergence for sparse variational gaussian process regression. In *International Conference on Machine Learning*, pages 862–871. PMLR, 2019.

Jie Chen and Michael Stein. Linear-cost covariance functions for gaussian random fields. *Journal of the American Statistical Association*, 2021. doi: 10.1080/01621459.2021.1919122.

Noel Cressie. *Statistics for spatial data*. John Wiley & Sons, 2015.

Timothy A Davis. *Direct Methods for Sparse Linear Systems*. SIAM, 2006.

Marc Peter Deisenroth, Dieter Fox, and Carl Edward Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE transactions on pattern analysis and machine intelligence*, 37(2):408–423, 2013.

Liang Ding, Rui Tuo, and Shahin Shahrampour. Generalization guarantees for sparse kernel approximation with entropic optimal features. In *Proceedings of the 37th International Conference on Machine Learning*, pages 2545–2555, 2020.

Donald W Fausett and Charles T Fulton. Large least squares problems involving kronecker products. *SIAM Journal on Matrix Analysis and Applications*, 15(1):219–227, 1994.

Jacob R Gardner, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. *arXiv preprint arXiv:1809.11165*, 2018.

Thomas Gerstner and Michael Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18(3):209–232, 1998.

Alexander Graham. *Kronecker Products and Matrix Calculus with Applications*. Courier Dover Publications, 2018.

Robert B Gramacy. *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. Chapman and Hall/CRC, 2020.

Robert B Gramacy and Daniel W Apley. Local gaussian process approximation for large computer experiments. *Journal of Computational and Graphical Statistics*, 24(2):561–578, 2015.

Jouni Hartikainen and Simo Särkkä. Kalman filtering and smoothing solutions to temporal gaussian process regression models. In *2010 IEEE international workshop on machine learning for signal processing*, pages 379–384. IEEE, 2010.

Philipp Hennig, Michael A Osborne, and Mark Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142, 2015.

Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *J. Glob. Optim.*, 13(4):455–492, 1998.

Emmanuel Kamgnia and Louis Bernard Nguenang. Some efficient methods for computing the determinant of large sparse matrices. *Revue Africaine de la Recherche en Informatique et Mathématiques Appliquées*, 17:73–92, 2014.

Matthias Katzfuss. A multi-resolution approximation for massive spatial datasets. *Journal of the American Statistical Association*, 112(517):201–214, Jan 2017.

Charles D Keeling and TP Whorf. Atmospheric carbon dioxide record from mauna loa. *Carbon Dioxide Research Group, Scripps Institution of Oceanography, University of California La Jolla, California*, pages 92093–0444, 2005.

Jackson Loper, David Blei, John P Cunningham, and Liam Paninski. A general linear-time inference method for Gaussian processes on one dimension. *Journal of Machine Learning Research*, 22(234):1–36, 2021.

Marcin Molga and Czesław Smutnicki. Test functions for optimization needs. *Test functions for optimization needs*, 101:48, 2005.

M Plumlee, CB Erickson, BE Ankenman, and E Lawrence. Composite grid designs for adaptive computer experiments with fast inference. *Biometrika*, 108(3):749–755, 2021.

Matthew Plumlee. Fast prediction of deterministic functions using sparse grid experimental designs. *Journal of the American Statistical Association*, 109(508):1581–1591, 2014.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems 20*, pages 1177–1184, 2007.

Carl Edward Rasmussen. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Carl Edward Rasmussen and Hannes Nickisch. Gaussian processes for machine learning (gpml) toolbox. *Journal of Machine Learning Research*, 11(100):3011–3015, 2010.

Yunus Saatçi. *Scalable inference for structured Gaussian process models*. PhD thesis, Citeseer, 2012.

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

Thomas J Santner, Brian J Williams, William I Notz, and Brain J Williams. *The Design and Analysis of Computer Experiments*, volume 1. Springer, 2003.

Simo Sarkka, Arno Solin, and Jouni Hartikainen. Spatiotemporal learning via infinite-dimensional bayesian filtering and smoothing: A look at gaussian process regression through kalman filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013.

Alex J Smola and Bernhard Schölkopf. Sparse greedy matrix approximation for machine learning. 2000.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

Bharath Sriperumbudur and Zoltan Szabo. Optimal rates for random fourier features. *Advances in Neural Information Processing Systems*, 28:1144–1152, 2015.

Elias M Stein and Rami Shakarchi. *Complex Analysis*. Princeton University Press, 2003.

Michael L Stein. *Interpolation of Spatial Data: some theory for kriging*. Springer Science & Business Media, 1999.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Artificial intelligence and statistics*, pages 567–574. PMLR, 2009.

Rui Tuo and Wenjia Wang. Kriging prediction with isotropic matern correlations: robustness and experimental designs. *Journal of Machine Learning Research*, 21(187):1–38, 2020.

Rui Tuo and C. F. Jeff Wu. A theoretical framework for calibration in computer models: parametrization, estimation and convergence properties. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):767–795, 2016.

Christopher Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*, pages 682–688, 2001.

Andrew Wilson and Hannes Nickisch. Kernel interpolation for scalable structured Gaussian processes (kiss-gp). In *International Conference on Machine Learning*, pages 1775–1784, 2015.

Andrew Gordon Wilson. *Covariance kernels for fast automatic pattern discovery and extrapolation with Gaussian processes*. PhD thesis, Citeseer, 2014.

Andrew TA Wood and Grace Chan. Simulation of stationary Gaussian processes in $[0,1]^d$. *Journal of Computational and Graphical Statistics*, 3(4):409–432, 1994.

Yunong Zhang, William E Leithead, and Douglas J Leith. Time-series gaussian process regression based on toeplitz computation of $O(N^2)$ operations and $O(N)$-level storage. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 3711–3716. IEEE, 2005.