Research Article

Alexander Litvinenko*, David Keyes, Venera Khoromskaia, Boris N. Khoromskij and Hermann G. Matthies

Tucker Tensor Analysis of Matérn Functions in Spatial Statistics

https://doi.org/10.1515/cmam-2018-0022
Received November 21, 2017; revised March 12, 2018; accepted May 2, 2018

Abstract: In this work, we describe advanced numerical tools for working with multivariate functions and for the analysis of large data sets. These tools will drastically reduce the required computing time and the storage cost, and, therefore, will allow us to consider much larger data sets or finer meshes. Covariance matrices are crucial in spatio-temporal statistical tasks, but are often very expensive to compute and store, especially in three dimensions. Therefore, we approximate covariance functions by cheap surrogates in a low-rank tensor format. We apply the Tucker and canonical tensor decompositions to a family of Matérn- and Slater-type functions with varying parameters and demonstrate numerically that their approximations exhibit exponentially fast convergence. We prove the exponential convergence of the Tucker and canonical approximations in tensor rank parameters. Several statistical operations are performed in this low-rank tensor format, including evaluating the conditional covariance matrix, spatially averaged estimation variance, computing a quadratic form, determinant, trace, loglikelihood, inverse, and Cholesky decomposition of a large covariance matrix. Low-rank tensor approximations reduce the computing and storage costs essentially. For example, the storage cost is reduced from an exponential $\mathcal{O}(n^d)$ to a linear scaling $\mathcal{O}(drn)$, where d is the spatial dimension, n is the number of mesh points in one direction, and r is the tensor rank. Prerequisites for applicability of the proposed techniques are the assumptions that the data, locations, and measurements lie on a tensor (axes-parallel) grid and that the covariance function depends on a distance, $\|x - y\|$.

Keywords: Fourier Transform, Low-Rank Tensor Approximation, Geostatistical Optimal Design, Kriging, Matérn Covariance, Hilbert Tensor, Kalman Filter, Bayesian Update, Loglikelihood Surrogate

MSC 2010: 60H15, 60H35, 65N25

1 Introduction

Nowadays it is very common to work with large spatial data sets [15, 44, 52, 61, 62, 64], for instance, with satellite data, collected over a very large area (e.g., the data collected by the National Center for Atmospheric Research, USA, https://www.earthsystemgrid.org/). This data can also come from a computer simulator code

***Corresponding author: Alexander Litvinenko,** Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology, Thuwal-Jeddah, Saudi Arabia, e-mail: alexander.litvinenko@kaust.edu.sa. http://orcid.org/0000-0001-5427-3598
David Keyes, Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology, Thuwal-Jeddah, Saudi Arabia, e-mail: david.keyes@kaust.edu.sa
Venera Khoromskaia, Max-Planck Institute for Mathematics in the Sciences, 04103 Leipzig; and Max-Planck Institute for Dynamics of Complex Technical Systems, 39106 Magdeburg, Germany, e-mail: vekh@mis.mpg.de
Boris N. Khoromskij, Max-Planck Institute for Mathematics in the Sciences, 04103 Leipzig, Germany, e-mail: bokh@mis.mpg.de
Hermann G. Matthies, Institute of Scientific Computing, TU Braunschweig, 38106 Braunschweig, Germany, e-mail: wire@tu-braunschweig.de

as a solution of a certain multiparametric equation (e.g., Weather research and Forecasting model, https://www.mmm.ucar.edu/weather-research-and-forecasting-model), it could be also sensor data from multiple sources. Typical operations in spatial statistics, such as evaluating the spatially averaged estimation variance, computing quadratic forms of the conditional covariance matrix, or computing maximum of likelihood function [62] require high computing power and time. Our motivation for using low-rank tensor techniques is that operations on advanced matrices, such as hierarchical, low-rank and sparse matrices, are limited by their high computational costs, especially in three dimensions and for a large number of observations.

A tensor can be simply defined as a high-order matrix, where multi-indices are used instead of indices (see Section 3 and equation (3.1) for a rigorous definition). One way to obtain a tensor from a vector or matrix is to reshape it. For example, we assume that $\mathbf{v} \in \mathbb{R}^{10^6}$ is a vector. We reshape it and obtain a matrix of size $10^3 \times 10^3$, or a tensor of order 3 of size $10^2 \times 10^2 \times 10^2$ or a tensor of order 6 of size $10 \times \cdots \times 10$ (6 times). Each element of such a six-dimensional hypercube is described by the multi-index $\alpha = (\alpha_1, \dots, \alpha_6)$. The obtained tensors contain not only rows and columns, but also *slices* and *fibers* [10, 40, 41]. These slices and fibers can be analyzed for linear dependences, super symmetry, or sparsity and may result in a strong data compression. Another difference between tensors and matrices is that a matrix (obtained, for instance, after the discretization of a kernel $c(\mathbf{x}, \mathbf{y}) = c(|\mathbf{x} - \mathbf{y}|)$) separates a point $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$ from a point $\mathbf{y} = (y_1, \dots, y_d) \in \mathbb{R}^d$, whereas the corresponding tensor (depending on tensor format) separates $x_1 - y_1$ from $x_2 - y_2, x_3 - y_3, \dots, x_d - y_d$. This implies that tensors may have not just one rank like a matrix, but many. Therefore, we speak about a *tensor rank*, but not a matrix rank. In this work, we consider two very common tensor formats: canonical (denoted as CP) and Tucker (see Section 3).

Low-rank tensor methods can be gainfully combined with other data-compression techniques in low dimensions. For example, a three-dimensional function can be approximated as the sum of the tensor products of one-dimensional function. Then the usual matrix techniques can be applied to those one-dimensional functions.

To be more concrete, we consider a relatively wide class of Matérn covariance functions. We demonstrate how to approximate Matérn covariance matrices in a low-rank tensor format, then how to perform typical Kriging and spatial statistics operations in this tensor format. Matérn covariance matrices typically depend on three to five unknown hyper-parameters, such as smoothness, three covariance lengths (in a three-dimensional anisotropic case), and variance. We study the dependences of tensor ranks and approximation errors on these parameters. Splitting the spatial variables via low-rank techniques reduces the computing cost for a matrix-vector product from $\mathcal{O}(N^2)$ to $\mathcal{O}(dr^2n^2)$ FLOPs, where d is the spatial dimension, r is the tensor rank, and n is the number of mesh points along the longest edge of the computational domain. For simplicity, we assume that $N = n^d$ (e.g., $d = 4$ for a time-space problem in three dimensions). Other motivating factors for applying low-rank tensor techniques include the following:

(1) The storage cost is reduced from $\mathcal{O}(n^d)$ to $\mathcal{O}(drn)$ or, depending on the tensor format, to $\mathcal{O}(drn + r^d)$, where $d > 1$.
(2) The low-rank tensor technique allows us to compute not only the matrix-vector product, but also the inverse $\mathbf{C}^{-1}$, square root $\mathbf{C}^{\frac{1}{2}}$, matrix exponent $\exp(\mathbf{C})$, $\operatorname{trace}(\mathbf{C})$, $\det(\mathbf{C})$, and a likelihood function.
(3) The low-rank tensor approximation is relatively new, but already a well-studied technique with free software libraries available.
(4) The approximation accuracy is fully controlled by the tensor rank. The full rank gives an exact representation.
(5) Low-rank techniques are either faster than a Fourier transform ($\mathcal{O}(drn)$ vs. $\mathcal{O}(n^d \log n^d)$) or can be efficiently combined with it [13, 51].

General limitations of the tensor technique are the following:
(a) It could be time consuming to compute a low-rank tensor decomposition.
(b) It requires axes-parallel mesh.
(c) Some theoretical estimations exist for functions depending on $|x - y|$ (although more general functions have a low-rank representation in practice).

During the last few years, there has been great interest in numerical methods for representing and approximating large covariance matrices [1, 2, 43, 44, 51, 54, 56]. Low-rank tensors were previously applied to accelerated Kriging and spatial design by orders of magnitude [51]. The covariance matrix under consideration was assumed to be circulant, and the first column had a low-rank decomposition. Therefore, d-dimensional Fourier was applied to and drastically reduce the storage and the computing cost.

The maximum likelihood estimator was computed for parameter-fitting given Gaussian observations with a Matérn covariance matrix [47]. The presented framework for unstructured observations in two spatial dimensions allowed for an evaluation of the log-likelihood and its gradient with computational complexity $\mathcal{O}(n^{\frac{3}{2}})$. The $\mathcal{H}$-matrix techniques [19, 21, 22] provide the efficient data sparse approximation for the differential and integral operators in $\mathbb{R}^d$, $d = 1, 2, 3$. $\mathcal{H}$-matrices are very robust for approximating the covariance matrix [1, 2, 4, 26, 38, 56], its inverse [1], and its Cholesky decomposition [38, 43, 44], but can also be expensive, especially for large n in three dimensions. Namely, the complexity in three dimensions will be $Ck^{d-1}N \log N$, where $N = n^d$, $d = 3$, $k \ll n$ is the rank and C is a large constant which scales exponentially in dimension d, see [22]. Thus, the $\mathcal{H}$-matrix techniques scale exponentially in dimension size. Therefore, more efficient methods for fast and efficient matrix linear algebra operations are still needed.

The key idea is to compute a low-rank decomposition not of the covariance function (it could be hard), but of its analytically known spectral density (which could be a much easier object) and then apply the inverse Fourier to the obtained low-rank components. The Fourier transformation of the Matérn covariance function is known analytically as the Hilbert tensor. This Hilbert tensor can be decomposed numerically in a low-rank tensor format. Both the Fourier transformation and its inverse have the canonical (CP) tensor rank-1. Therefore, the inverse Fourier does not change the tensor rank of the argument. By applying the inverse Fourier to the low-rank tensor, we obtain a low-rank approximation of the initial covariance matrix, which can be further used in the Kalman filter update, Karhunen–Loève expansion, Bayesian update, and Kriging.

The structure of the paper is as follows. In Section 2, we list typical tasks from statistics that motivate us to use low-rank tensor techniques and define the Matérn covariance functions and their Fourier transformations. Section 3 is devoted to low-rank tensor decomposition. Sections 3.4, 3.5 and 3.6 contain the main theoretical contribution of this work. We present low-rank tensor techniques and separate radial basis functions using the Laplace transform and the sinc quadrature, give estimations of the approximation error, convergence rate, and the tensor rank, and we also prove the existence of a low-rank approximation of a Matérn function. Section 4 contains another important contribution of this work, namely, the solutions to typical statistical tasks in the low-rank tensor format.

2 Motivation

2.1 Problem Settings in Spatial Statistics

Below, we formulate *five tasks*. These computational tasks are very common and important in statistics. Fast and efficient solution of these tasks will help to solve many real-world problems, such as the weather prediction, moisture modeling, and optimal design in geostatistics.

Task 1: Approximate a Matérn covariance function in a low-rank tensor format. The covariance function $c(\mathbf{x}, \mathbf{y})$, $\mathbf{x} = (x_1, \dots, x_d)$, $\mathbf{y} = (y_1, \dots, y_d)$, is discretized on a tensor grid with N mesh points, $N = n^d$, $d \geq 1$ and $\varepsilon > 0$. The task is to find the following decomposition into one-dimensional functions:

$$\left\| c(\mathbf{x}, \mathbf{y}) - \sum_{i=1}^{r} \prod_{\mu=1}^{d} c_{i\mu}(x_\mu, y_\mu) \right\| \leq \varepsilon$$

for some given $\varepsilon > 0$. Alternatively, we may look for factors $\mathbf{C}_{i\mu}$ such that

$$\left\| \mathbf{C} - \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu} \right\| \leq \varepsilon.$$

Here, the matrices $\mathbf{C}_{i\mu}$ correspond to the one-dimensional covariance functions $c_{i\mu}(x_\mu, y_\mu)$ in the direction μ.

Task 2: Computing of square root of C. The square root $\mathbf{C}^{\frac{1}{2}}$ of the covariance matrix $\mathbf{C}$ is needed in order to generate random fields and processes. It is also used in the Kalman filter update.

Task 3: Kriging. Spatial statistics and Kriging [39] are used to model the distribution of ore grade, forecast of rainfall intensities, moisture, temperatures, or contaminant. The missing values are interpolated from the known measurements by Kriging [30, 46]. When considering space-time Kriging on fine meshes [9, 14, 28, 66], Kriging may easily exceed the computational power of modern computers. Estimating the variance of Kriging and geostatistical optimal design problems are especially numerically intensive [48, 50, 60].

The Kriging can be defined as follows. Let $\hat{\mathbf{s}}$ be the $N \times 1$ vector of values to be estimated with zero expectation and covariance matrix $\mathbf{C}_{ss}$. Let $\mathbf{y}$ be the $m \times 1$, $m \ll N$, vector of measurements. The corresponding cross- and auto-covariance matrices are denoted by $\mathbf{C}_{sy}$ and $\mathbf{C}_{yy}$, and sized $N \times m$ and $m \times m$, respectively. If the measurements are subject to error, an error covariance matrix $\mathbf{R}$ is included in $\mathbf{C}_{yy}$. Using this notation, the Kriging estimate $\hat{\mathbf{s}}$ is given by $\hat{\mathbf{s}} = \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{y}$.

Task 4: Geostatistical design. The goal of geostatistical optimal design is to optimize the sampling patterns from which the data values in $\mathbf{y}$ will be obtained. The objective function that will be minimized is typically a scalar measure of either the conditional covariance matrix or the estimation variance (4.4). The two most common measures for geostatistical optimal design are φ_A and φ_C:

$$\varphi_A = N^{-1}\operatorname{trace}[\mathbf{C}_{ss|y}] \quad \text{and} \quad \varphi_C = \mathbf{z}^T(\mathbf{C}_{ss|y})\mathbf{z}, \tag{2.1}$$

where $\mathbf{C}_{ss|y} := \mathbf{C}_{ss} - \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys}$, see [48, 50].

Task 5: Computing the joint Gaussian log-likelihood function. We assume that $\boldsymbol{z} \in \mathbb{R}^N$ is an available vector of measurements, and $\boldsymbol{\theta}$ is an unknown vector of the parameters of a covariance matrix $\mathbf{C}$. The task is to compute the maximum likelihood estimation (MLE), where the log-likelihood function is as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log\det\{\mathbf{C}(\boldsymbol{\theta})\} - \frac{1}{2}(\mathbf{z}^T \cdot \mathbf{C}(\boldsymbol{\theta})^{-1}\mathbf{z}).$$

The difficulty here is that each iteration step of a maximization procedure requires the solution of a linear system $\mathbf{Lv} = \mathbf{z}$, the Cholesky decomposition, and the determinant.

In Section 4 we give detailed solutions. We give strict definition of tensors later in Section 3.

2.2 Matérn Covariance and Its Fourier Transform

A low-rank approximation of the covariance function is a key component of the tasks formulated above. Among of the many covariance models available, the Matérn family [25, 45] is widely used in spatial statistics, geostatistics [7], machine learning [4], image analysis, weather forecast, moisture modeling, and as the correlation for temperature fields [49]. The work [25] introduced the Matérn form of spatial correlations into statistics as a flexible parametric class with one parameter determining the smoothness of the underlying spatial random field.

The main idea of this low-rank approximation is shown in Figure 1 and explained in details in Section 3.3. Figure 1 demonstrates two possible ways to find a low-rank tensor approximation of the Matérn covariance function. The first way (marked with "?") is not so trivial and the second via the Fast Fourier Transform (FFT), low-rank and the inverse FFT (IFFT) is more trivial. We use here the fact that the FT of the Matérn covariance is analytically known and has a known low-rank approximation. The IFFT can be computed numerically and does not change the tensor ranks.

The Matérn covariance function is defined as

$$C_{\nu,\ell}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu}\mathcal{K}_\nu\left(\frac{\sqrt{2\nu}r}{\ell}\right), \tag{2.2}$$

where distance $r := \|x - y\|$, x, y two points in $\mathbb{R}^d$, $\nu > 0$ defines the smoothness of the random field, and $\mathcal{K}_\nu$ denotes the modified Bessel function of order ν. The larger ν, the smoother the random field. The parameter $\ell > 0$ is a spatial range parameter that measures how quickly the correlation of the random field decays with

$$\begin{tikzcd}
C_{\nu,\ell}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}r}{\ell}\right) \arrow[r, "?"] \arrow[d, "\text{FFT}"] & \approx \sum_{i=1}^{R} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu} \\
f_{\alpha,\nu}(\rho) := \frac{\Gamma(\nu+\frac{d}{2})\alpha^{2\nu}}{\pi^{\frac{d}{2}}\Gamma(\nu)} \frac{1}{(\alpha^2+\rho^2)^{\nu+\frac{d}{2}}} \arrow[r, "\text{Low-rank}", "\text{approximation}"'] & \approx \sum_{i=1}^{R} \bigotimes_{\nu=1}^{d} \tilde{u}_{i\nu} \arrow[u, "\text{IFFT}"']
\end{tikzcd}$$

Figure 1: Two possible ways to find a low rank tensor approximation of the Matérn covariance matrix $C_{\nu,\ell}(r)$.

distance, with larger ℓ corresponding to a slower decay (keeping ν fixed). When $\nu = \frac{1}{2}$ (see [55]), the Matérn covariance function reduces to the exponential covariance model and describes a rough field. The value $\nu = \infty$ corresponds to a Gaussian covariance model, which describes a very smooth field, that is infinitely differentiable. Random fields with a Matérn covariance function are $\lfloor \nu - 1 \rfloor$ times mean square differentiable. Thus, the hyperparameter ν controls the degree of smoothness.

The d-dimensional Fourier transform $\boldsymbol{F}^d(C(r,\nu)$ of the Matérn kernel, defined in equation (2.2), in $\mathbb{R}^d$ is given by [45]

$$\boldsymbol{U}(\xi) := \boldsymbol{F}^d(C(r,\nu) = \beta \cdot \left(1 + \frac{\ell^2}{2\nu}|\xi|^2\right)^{-\nu-\frac{d}{2}}, \tag{2.3}$$

where $\beta = \beta(\nu, \ell, n)$ is a constant and $|\xi|$ is the Euclidean distance in $\mathbb{R}^d$. The following tensor approach also applies to the case of anisotropic distance, where $r^2 = \sqrt{\langle A(x-y), (x-y)\rangle}$, and A is a positive diagonal $d \times d$ matrix.

3 Low-Rank Tensor Decompositions

In this section, we review the definitions of the CP and Tucker tensor formats. Then we provide the analytic sinc-based proof of the existence of low-rank tensor approximations of Matérn functions. We investigate numerically the behavior of the Tucker and CP ranks across a wide range of parameters specific to the family of Matérn kernels in equation (2.3). The Tucker tensor format is used in this work for additional rank compression of the CP factors. There are no reliable algorithms to compute CP decomposition, which can be difficult to compute, but there are such algorithms for Tucker decomposition. The Tucker decomposition is only limited with respect to the available memory storage, since the term r^d in $\mathcal{O}(drn + r^d)$ grows exponentially with d.

3.1 General Definitions

CP and Tucker rank-structured tensor formats have been applied for the quantitative analysis of correlation in multidimensional experimental data for a long time in chemometrics and signal processing [8, 59]. The Tucker tensor format was introduced in 1966 for tensor decomposition of multidimensional arrays in chemometrics [65]. Though the canonical representation of multivariate functions was introduced as early as in 1927 [29], only the Tucker tensor format provides a stable algorithm for decomposition of full-size tensors. A mathematical approval of the Tucker decomposition algorithm was presented in papers on higher-order singular value decomposition (HOSVD) and the Tucker ALS algorithm for orthogonal Tucker approximation of higher-order tensors [10]. For higher dimensions, the so-called Matrix Product States (MPS) (see the survey paper [57]) or the Tensor Train (TT) [53] decompositions can be applied. However, for three-dimensional applications, the Tucker and CP tensor formats remain the best choices. The fast convergence of the Tucker decomposition was proved and demonstrated numerically for higher-order tensors that arise from the discretization of linear operators and functions in $\mathbb{R}^d$ for a class of function-related tensors and Green's kernels in particular, it was found that the approximation error of the Tucker decomposition decayed exponentially in the Tucker rank [33, 36].

These results inspired the canonical-to-Tucker (C2T) and Tucker-to-canonical (T2C) decompositions for function-related tensors in the case of large input ranks, as well as the multigrid Tucker approximation [37].

A tensor of order d in a full format is defined as a multidimensional array over a d-tuple index set:

$$\mathbf{A} = [a_{i_1,\dots,i_d}] \equiv [a(i_1,\dots,i_d)] \ \in \mathbb{R}^{n_1\times\cdots\times n_d} \quad \text{with } i_\ell \in I_\ell := \{1,\dots,n_\ell\}. \tag{3.1}$$

Here, $\mathbf{A}$ is an element of the linear space

$$\mathbb{V}_n = \bigotimes_{\ell=1}^{d} \mathbb{V}_\ell, \quad \mathbb{V}_\ell = \mathbb{R}^{I_\ell}$$

equipped with the Euclidean scalar product $\langle\cdot,\cdot\rangle : \mathbb{V}_n \times \mathbb{V}_n \to \mathbb{R}$, defined as

$$\langle \mathbf{A}, \mathbf{B}\rangle := \sum_{(i_1\dots i_d)\in\mathcal{I}} a_{i_1\dots i_d} b_{i_1\dots i_d} \quad \text{for } \mathbf{A}, \mathbf{B} \in \mathbb{V}_n.$$

Tensors with all dimensions having equal size $n_\ell = n$, $\ell = 1,\dots,d$, are called $n^{\otimes d}$ tensors. The required storage size scales exponentially with the dimension, n^d, which results in the so-called "curse of dimensionality".

To avoid exponential scaling in the dimension, the rank-structured separable representations (approximations) of the multidimensional tensors can be used. The simplest separable element is given by the rank-1 tensor

$$\mathbf{U} = \mathbf{u}^{(1)} \otimes \cdots \otimes \mathbf{u}^{(d)} \in \mathbb{R}^{n_1\times\cdots\times n_d},$$

with entries $u_{i_1,\dots,i_d} = u^{(1)}_{i_1}\cdots u^{(d)}_{i_d}$, which requires only $n_1 + \cdots + n_d$ numbers for storage.

The rank-1 canonical tensor is a discrete counterpart of the separable d-variate function, which can be represented as the product of univariate functions

$$f(x_1, x_2, \dots, x_d) = f_1(x_1)f_2(x_2)\dots f_d(x_d).$$

An example of the separable d-variate function is $f(x_1,x_2,x_3) = e^{(x_1+x_2+x_3)}$. Then, by discretization of this multivariate function on a tensor grid in a computational box, we obtain a canonical rank-1 tensor.

A tensor in the R-term canonical format is defined by a finite sum of rank-1 tensors (Figure 2, left)

$$\mathbf{A}_c = \sum\nolimits_{k=1}^{R} \xi_k \mathbf{u}^{(1)}_k \otimes \cdots \otimes \mathbf{u}^{(d)}_k, \quad \xi_k \in \mathbb{R}, \tag{3.2}$$

where $\mathbf{u}^{(\ell)}_k \in \mathbb{R}^{n_\ell}$ are normalized vectors, and R is the canonical rank. The storage cost of this parametrization is bounded by dRn. An element $a(i_1,\dots,i_d)$ of the tensor $\mathbf{A} = \sum_{i=1}^{R}\bigotimes_{\nu=1}^{d} u_{i\nu}$ can be computed as

$$a(i_1,\dots,i_d) = \sum_{\alpha=1}^{R} u_1(i_1,\alpha)u_2(i_2,\alpha)\dots u_d(i_d,\alpha).$$

An alternative (contracted product) notation is used in computer science community:

$$\mathbf{A} = \mathbf{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 \cdots \times_d U^{(d)}, \tag{3.3}$$

where $\mathbf{C} = \operatorname{diag}\{c_1,\dots,c_d\} \in \mathbb{R}^{R^{\otimes d}}$, and $U^{(\ell)} = [\mathbf{u}^{(\ell)}_1 \dots \mathbf{u}^{(\ell)}_R] \in \mathbb{R}^{n_\ell\times R}$. An analogous multivariate function can be represented by a sum of univariate functions

$$f(x_1, x_2, \dots, x_d) = \sum_{k=1}^{R} f_{1,k}(x_1)f_{2,k}(x_2)\dots f_{d,k}(x_2).$$

For $d \geq 3$, there are no stable algorithms to compute the canonical rank of a tensor $\mathbf{A}$, that is, the minimal number R in representation (3.2), and the respective decomposition with the polynomial cost in d, i.e., the computation of the canonical decomposition is an N-P hard problem [27].

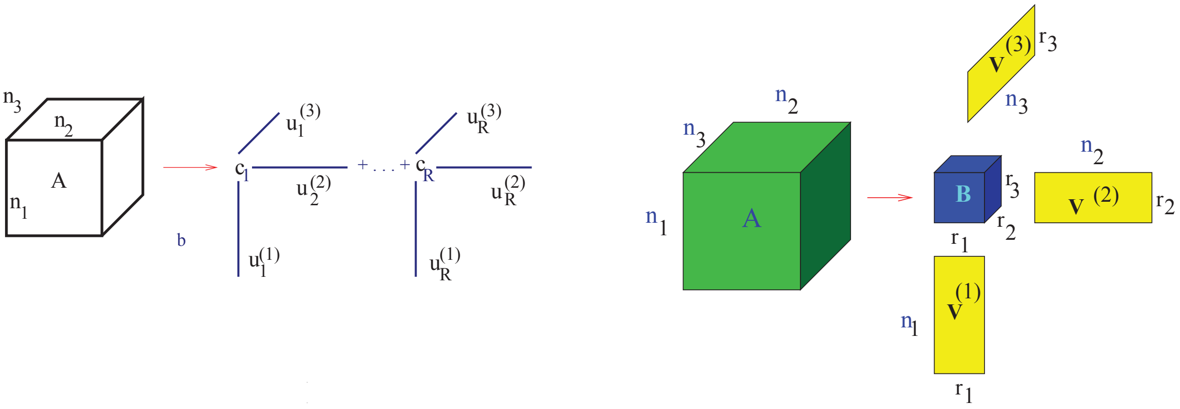

Figure 2: Canonical (left) and Tucker (right) decompositions of three-dimensional tensors.

The Tucker tensor format (Figure 2, right) is suitable for stable numerical decompositions with a fixed truncation threshold. We say that the tensor $\mathbf{A}$ is represented in the rank-$\mathbf{r}$ orthogonal Tucker format with the rank parameter $\mathbf{r} = (r_1, \ldots, r_d)$ if

$$\mathbf{A} = \sum_{\nu_1=1}^{r_1} \cdots \sum_{\nu_d=1}^{r_d} \beta_{\nu_1,\ldots,\nu_d} \mathbf{v}_{\nu_1}^{(1)} \otimes \cdots \otimes \mathbf{v}_{\nu_\ell}^{(\ell)} \cdots \otimes \mathbf{v}_{\nu_d}^{(d)}, \quad \ell = 1, \ldots, d,$$

where $\{\mathbf{v}_{\nu_\ell}^{(\ell)}\}_{\nu_\ell=1}^{r_\ell} \in \mathbb{R}^{n_\ell}$ represents a set of orthonormal vectors for $\ell = 1, \ldots, d$, and $\mathbf{B} = [\mathbf{B}_{\nu_1,\ldots,\nu_d}] \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ is the Tucker core tensor. The storage cost for the Tucker tensor is bounded by $drn + r^d$, with $r = |\mathbf{r}| := \max_\ell r_\ell$. Using the orthogonal side matrices $V^{(\ell)} = [v_1^{(\ell)} \ldots v_{r_\ell}^{(\ell)}]$ and contracted products, the Tucker tensor decomposition may be presented in the alternative notation

$$\mathbf{A}_{(\mathbf{r})} = \mathbf{B} \times_1 V^{(1)} \times_2 V^{(2)} \times_3 \cdots \times_d V^{(d)}.$$

In the case $d = 2$, the orthogonal Tucker decomposition is equivalent to the singular value decomposition (SVD) of a rectangular matrix.

3.2 Tucker Decomposition of Full Format Tensors

We use the following algorithm to compute the Tucker decomposition of the full format tensor. The most time-consuming part of the Tucker algorithm is higher-order singular value decomposition (HOSVD), the computation of the initial guess for matrices $V^{(\ell)}$ using the SVD of the matrix unfolding $A_{(\ell)}$, $\ell = 1, 2, 3$ (Figure 3), of the original tensor along each mode of a tensor [10]. Figure 3 illustrates the matrix unfolding of the full format tensor $\mathbf{A}$ (see (3.1)) along the index set $I_1 = \{1, \ldots, n_1\}$.

The second part of the algorithm is the ALS procedure. For every tensor mode, a "single-hole" tensor of reduced size is constructed by the mapping all of the modes of the original tensor except one into the subspaces $V^{(\ell)}$. Then the subspace $V^{(\ell)}$ for the current mode is updated by SVD of the unfolding of the "single

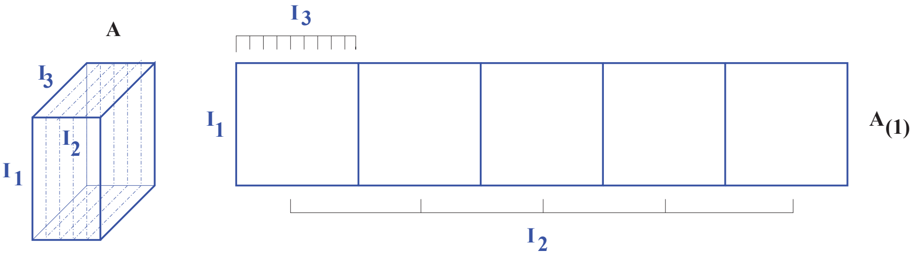

Figure 3: Unfolding of a three-dimensional tensor along the mode I_ℓ with $\ell = 1$.

hole" tensor for this mode. This alternates over all modes of the tensor, which are updated at the current iteration of ALS. The final step of the algorithm is computation of the core tensor by using the ultimate mapping matrices from ALS.

The numerical cost of Tucker decomposition for full size tensors is dominated by the initial guess, which is estimated as $O(n^{d+1})$ when all $n_\ell = n$ are equal , or $O(n^4)$ for our three-dimensional case. This step restricts the available size of the tensor to be decomposed since, for conventional computers, the three-dimensional case $n_\ell > 10^2$ is the limiting case for SVD.

The multigrid Tucker algorithm for full size tensors allows the computational complexity to be linear in the full size of the tensor, $O(n^d)$, i.e., $O(n^3)$ for three-dimensional tensors [37]. It is computed on a sequence of diadically refined grids and is based on implementing the HOSVD only at the coarsest grid level, $n_0 \ll n$. The initial guess for the ALS procedure is computed at each refined level by the interpolation of the dominating Tucker subspaces obtained from the previous coarser grid. In this way, at fine three-dimensional Cartesian grids, we need 29 only $O(n^3)$ storage (to represent the initial full format tensor) to contract with the Tucker side matrices, obtained by the Tucker approximation via ALS on the previous grids.

3.3 Illustration of the Low-Rank Approximation Idea

In this subsection we describe a possible ways to find a low-rank tensor approximation of the Matérn covariance matrix $C(r,\nu)$ (Figure 1). Let $\boldsymbol{F}^d = \bigotimes_{\nu=1}^d \boldsymbol{F}_\nu$ be the d-dimensional Fourier transform, where $\boldsymbol{F}^{-d} = \bigotimes_{\nu=1}^d \boldsymbol{F}_\nu^{-1}$ is its inverse and $\bigotimes$ denotes the Kronecker product. We assume that $\boldsymbol{U}(\xi) = \boldsymbol{F}^d(C(r,\nu))$ is known analytically and has a low-rank tensor approximation $\mathbf{U} = \sum_{j=1}^r \bigotimes_{\nu=1}^d \mathbf{u}_{j\nu}$. Since the Fourier and inverse Fourier transformations do not change the Kronecker tensor rank of the argument [51], by applying the inverse Fourier, we obtain a low-rank representation of the covariance function by applying the inverse Fourier:

$$\boldsymbol{F}^{-d}(\mathbf{U}) = \Big(\bigotimes_{\nu=1}^d \boldsymbol{F}_\nu^{-1}\Big)\sum_{i=1}^r\Big(\bigotimes_{\nu=1}^d \mathbf{u}_{\nu i}\Big) = \sum_{i=1}^r \bigotimes_{\nu=1}^d (\boldsymbol{F}_\nu^{-1}(\mathbf{u}_{\nu i})) = \sum_{i=1}^r \bigotimes_{\nu=1}^d \tilde{\mathbf{u}}_{\nu i} =: C(r,\nu),$$

where $\tilde{\mathbf{u}}_{\nu i} := \boldsymbol{F}_\nu^{-1}(\mathbf{u}_{\nu i})$.

3.4 Sinc Approximation of the Matérn Function

The Sinc method provides a constructive approximation of the multivariate functions in the form of a low-rank canonical representation. It can be also used for the theoretical proof and for the rank estimation. Methods for the separable approximation of the three-dimensional Newton kernel and many other spherically symmetric functions that use the Gaussian sums have been developed since the initial studies in chemical [5] and mathematical literature [6, 17, 23, 63]. Here, we use a tensor-decomposition approach for lattice-structured interaction potentials [32]. We also recall the grid-based method for a low-rank canonical representation of the spherically symmetric kernel function $q(\|x\|)$, where $x \in \mathbb{R}^d$, $d = 2, 3, \dots$, by its projection onto the set of piecewise constant basis functions; see [3] for the case of Newton and Yukawa kernels for $x \in \mathbb{R}^3$.

Following the standard schemes, we introduce the uniform $n \times n \times n$ rectangular Cartesian grid Ω_n with mesh size $h = \frac{2b}{n}$ (we assume that n is even) in the computational domain $\Omega = [-b, b]^3$. Let $\{\psi_\mathbf{i}\}$ be a set of tensor-product piecewise constant basis functions

$$\psi_\mathbf{i}(\mathbf{x}) = \prod_{\ell=1}^3 \psi_{i_\ell}^{(\ell)}(x_\ell)$$

for the 3-tuple index $\mathbf{i} = (i_1, i_2, i_3) \in \mathcal{I}$, $\mathcal{I} = I_1 \times I_2 \times I_3$, with $i_\ell \in I_\ell = \{1, \dots, n\}$, where $\ell = 1,\ 2,\ 3$. The generating kernel $q(\|x\|)$ is discretized by its projection onto the basis set $\{\psi_\mathbf{i}\}$ in the form of a third-order tensor of size $n \times n \times n$, which is defined entry-wise as

$$\mathbf{Q} := [q_\mathbf{i}] \in \mathbb{R}^{n\times n\times n}, \quad q_\mathbf{i} = \int_{\mathbb{R}^3} \psi_\mathbf{i}(x) q(\|x\|)\,\mathrm{d}x. \tag{3.4}$$

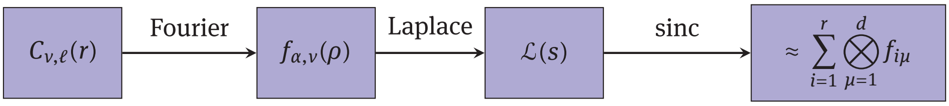

Figure 4: Scheme of the proof of existence of low-rank tensor approximation, $r = 2M + 1$.

The low-rank canonical decomposition of the third-order tensor $\mathbf{Q}$ is based on applying exponentially convergent sinc-quadratures to the integral representation of the function $q(p)$, $p \in \mathbb{R}$, in the form

$$q(p) = \int_{\mathbb{R}} a_1(t) e^{-p^2 a_2(t)}\, \mathrm{d}t,$$

specified by the weights $a_1(t)$, $a_2(t) > 0$. Figure 4 illustrates a scheme of the proof of existence of the canonical low-rank tensor approximation. It could be easier to apply the Laplace transform to the Fourier transform of a Matérn covariance matrix than to the Matérn covariance. To approximate the resulting Laplace integral we apply the sinc quadrature. The number $(2M + 1)$ of terms in the approximate sum (3.5) is the canonical tensor rank.

In particular, the sinc-quadrature for the Laplace–Gauss transform

$$q(p) = \int_{\mathbb{R}_+} a(t) e^{-t^2 p^2}\, \mathrm{d}t \approx \sum_{k=-M}^{M} a_k e^{-t_k^2 p^2} \quad \text{for } |p| > 0,\ p \in \mathbb{R} \tag{3.5}$$

can be applied, where the quadrature points (t_k) and weights (a_k) are given by

$$t_k = k\hbar_M \quad \text{and} \quad a_k = a(t_k)\hbar_M, \quad \text{where } \hbar_M = C_0 \frac{\log(M)}{M},\ C_0 > 0. \tag{3.6}$$

Under the assumption $0 < a_0 \le |p| < \infty$, this quadrature can be proven to provide an exponential convergence rate in M (uniformly in p) for a class of functions $a(z)$ that are analytic in a certain strip $|z| \le D$ of the complex plane such that the functions $a_1(t)e^{-p^2 a_2(t)}$ decay polynomially or exponentially on the real axis. The exponential convergence of the sinc-approximation in the number of terms (i.e., the canonical rank $R = 2M + 1$) was analyzed elsewhere [6, 23, 63].

We assume that a representation similar to (3.5) exists for any fixed $x = (x_1, x_2, x_3) \in \mathbb{R}^3$ such that $\|x\| > a_0 > 0$. Then we apply the sinc-quadrature approximation (3.5) and (3.6) to obtain the separable expansion

$$q(\|x\|) = \int_{\mathbb{R}_+} a(t) e^{-t^2\|x\|^2}\, \mathrm{d}t \approx \sum_{k=-M}^{M} a_k e^{-t_k^2\|x\|^2} = \sum_{k=-M}^{M} a_k \prod_{\ell=1}^{3} e^{-t_k^2 x_\ell^2}, \tag{3.7}$$

providing an exponential convergence rate in M:

$$\left| q(\|x\|) - \sum_{k=-M}^{M} a_k e^{-t_k^2\|x\|^2} \right| \le \frac{C}{a} e^{-\beta\sqrt{M}} \quad \text{with some } C, \beta > 0.$$

By combining (3.4) and (3.7), and taking into account the separability of the Gaussian basis functions, we arrive at the low-rank approximation of each entry of the tensor $\mathbf{Q}$:

$$q_{\mathbf{i}} \approx \sum_{k=-M}^{M} a_k \int_{\mathbb{R}^3} \psi_{\mathbf{i}}(x) e^{-t_k^2\|x\|^2}\, \mathrm{d}x = \sum_{k=-M}^{M} a_k \prod_{\ell=1}^{3} \int_{\mathbb{R}} \psi_{i_\ell}^{(\ell)}(x_\ell) e^{-t_k^2 x_\ell^2}\, \mathrm{d}x_\ell.$$

Recalling that $a_k > 0$, we define the vector $\mathbf{q}$ as

$$\mathbf{q}_k^{(\ell)} = a_k^{\frac{1}{3}} [b_{i_\ell}^{(\ell)}(t_k)]_{i_\ell=1}^{n_\ell} \in \mathbb{R}^{n_\ell} \quad \text{with } b_{i_\ell}^{(\ell)}(t_k) = \int_{\mathbb{R}} \psi_{i_\ell}^{(\ell)}(x_\ell) e^{-t_k^2 x_\ell^2}\, \mathrm{d}x_\ell.$$

Then the third-order tensor $\mathbf{Q}$ can be approximated by the R-term $(R = 2M + 1)$ canonical representation

$$\mathbf{Q} \approx \mathbf{Q}_R = \sum_{k=-M}^{M} a_k \bigotimes_{\ell=1}^{3} \mathbf{b}^{(\ell)}(t_k) = \sum_{k=-M}^{M} \mathbf{q}_k^{(1)} \otimes \mathbf{q}_k^{(2)} \otimes \mathbf{q}_k^{(3)} \in \mathbb{R}^{n \times n \times n}, \tag{3.8}$$

where $\mathbf{q}_k^{(\ell)} \in \mathbb{R}^n$. Given a threshold $\varepsilon > 0$, M can be chosen as the minimal number such that in the max-norm

$$\|\mathbf{Q} - \mathbf{Q}_R\| \leq \varepsilon \|\mathbf{Q}\|.$$

The skeleton vectors can be reindex by $k \mapsto k' = k + M + 1$, $\mathbf{q}_k^{(\ell)} \mapsto \mathbf{q}_{k'}^{(\ell)}$ $(k' = 1, \dots, R = 2M + 1)$, $\ell = 1, 2, 3$. The symmetric canonical tensor $\mathbf{Q}_R \in \mathbb{R}^{n \times n \times n}$ in (3.8) approximates the three-dimensional symmetric kernel function $q(\|x\|)$ $(x \in \Omega)$, centered at the origin, such that $\mathbf{q}_{k'}^{(1)} = \mathbf{q}_{k'}^{(2)} = \mathbf{q}_{k'}^{(3)}$ $(k' = 1, \dots, R)$.

In some applications, the tensor can be given in the canonical tensor format, but with large rank R and discretized on large grids $n \times n \times \cdots \times n$; thus, computation of the initial of guess in the Tucker-ALS decomposition algorithm becomes intractable. This situation may arise when composing the tensor approximation of complicated kernel functions from simple radial functions that can be represented in the low-rank CP format.

For such cases, the canonical-to-Tucker decomposition algorithm was introduced [37]. It is based on the minimization ALS procedure, similar to the Tucker algorithm, described in Section 3.2 for full-size tensors, but the initial guess is computed by just the SVD of the side matrices $U^{(\ell)} = [\mathbf{u}_1^{(\ell)} \dots \mathbf{u}_R^{(\ell)}] \in \mathbb{R}^{n_\ell \times R}$, $\ell = 1, 2, 3$; see (3.3). This schema is the Reduced HOSVD (RHOSVD), which does not require unfolding of the full tensor.

Another efficient rank-structured representation of the multidimensional tensors is the mixed-tensor format [31], which combines either the canonical-to-Tucker decomposition with the Tucker-to-canonical decomposition, or standard Tucker decomposition with the canonical-to-Tucker decomposition, in order to produce a canonical tensor from a full-size tensor.

3.5 Laplace Transform of the Covariance Matrix

The integral representations like (3.5) can be derived by the Laplace transform either directly to the Matérn covariance function or to its spectral density (2.3). For example, in the case of the Newton kernel, $q(p) = \frac{1}{p}$, and the Laplace–Gauss transform representation takes the form

$$\frac{1}{p} = \frac{2}{\sqrt{\pi}} \int_{\mathbb{R}_+} e^{-p^2 t^2} \, \mathrm{d}t, \quad \text{where } p = \|x\| = \sqrt{x_1^2 + x_2^2 + x_3^2}.$$

In this case, $\mathbf{q}_k^{(\ell)} = \mathbf{q}_{-k}^{(\ell)}$, and the sum of (3.8) reduces to $k = 0, 1, \dots, M$, implying that $R = M + 1$. Therefore, the Laplace transform representation of the Slater function $q(p) = e^{-2\sqrt{\alpha p}}$ (i.e., exponential covariance) with $p = \|x\|^2$ can be written as

$$q(p) = e^{-2\sqrt{\alpha p}} = \frac{\sqrt{\alpha}}{\sqrt{\pi}} \int_{\mathbb{R}_+} t^{-\frac{3}{2}} \exp\Big(-\frac{\alpha}{t} - pt\Big) \, \mathrm{d}t.$$

When the Matérn spectral density in (2.3) has an even dimension parameter $d = 2d_1$, $d_1 = 1, 2, \dots$ and $\nu = 0, 1, 2, \dots$, the Laplace transform

$$\frac{\eta!}{(p + a)^{\eta + 1}} = \int_{\mathbb{R}_+} t^\eta e^{-at} e^{-pt} \, \mathrm{d}t$$

can be applied after substituting $p = \|\xi\|^2$ in

$$q(p) = \beta\Big(1 + \frac{\ell^2}{2\nu} p\Big)^\eta, \quad \eta = -\nu - d_1.$$

If $-\eta = \nu + d_1 = \frac{1}{2}, \frac{3}{2}, \frac{5}{2}, \dots, \frac{2k+1}{2}, \dots$, then the Laplace transform is

$$\frac{(2\eta)! \sqrt{\pi}}{\eta! 4^\eta} \frac{1}{(p + a)^\eta \sqrt{p + a}} = \int_{\mathbb{R}_+} \frac{t^\eta}{\sqrt{t}} e^{-at} e^{-pt} \, \mathrm{d}t, \quad \eta \in \mathbb{N}.$$

3.6 Covariance Matrix in Rank-Structured Tensor Format

In what follows, we consider the CP approximation of the radial function $q(r)$ in the positive sector, i.e., on the domain $[0, b]^3$ (by symmetry, the canonical tensor can be extended to the whole computational domain $[-b, b]^3$.) Let the covariance function $q(r) = C_{\nu,\ell}(r)$ in (2.2) be represented by the rank-R symmetric CP tensor on an $n \times n \times n$ tensor grid, denoted by $\Omega_n \subset \Omega = [0, b]^3$ as described in the previous sections,

$$q(r) \mapsto \mathbb{Q} \approx \mathbb{Q}_R = \sum_{k=1}^{R} \mathbf{q}_k^{(1)} \otimes \mathbf{q}_k^{(2)} \otimes \mathbf{q}_k^{(3)} \in \mathbb{R}^{n\times n\times n} \tag{3.9}$$

with the same skeleton vectors $\mathbf{q}_k^{(\ell)} \in \mathbb{R}^n$ for $\ell = 1, 2, 3$.

We define the covariance matrix $\mathbf{C} = [c_{\mathbf{i},\mathbf{j}}] \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$ entry-wise by

$$c_{\mathbf{i},\mathbf{j}} = C_{\nu,\ell}(\|x_{\mathbf{i}} - y_{\mathbf{j}}\|), \quad \mathbf{i}, \mathbf{j} \in \mathcal{I}.$$

Using the tensor representation (3.9), we represent the large $n^3 \times n^3$ matrix $\mathbf{C}$ in the rank-R Kronecker (tensor) format as

$$\mathbf{C} \approx \mathbf{C}_R = \sum_{k=1}^{R} \mathbf{Q}_k^{(1)} \otimes \mathbf{Q}_k^{(2)} \otimes \mathbf{Q}_k^{(3)}, \tag{3.10}$$

where the symmetric Toeplitz matrix $\mathbf{Q}_k^{(\ell)} = \mathrm{Toepl}[\mathbf{q}_k^{(\ell)}] \in \mathbb{R}^{n\times n}$, $\ell = 1, 2, 3$, is defined by its first column, which is specified by the skeleton vectors $\mathbf{q}_k^{(1)} = \mathbf{q}_k^{(2)} = \mathbf{q}_k^{(3)}$ in the decomposition (3.9).

Figure 5 illustrates eight selected canonical generating vectors from $\mathbf{q}_k^{(1)}$ for $k = 1, \dots, R$, $R = 34$, on a grid of size $n = 2{,}049$ for the Slater function $e^{-\|x\|^p}$, which defines the corresponding Toeplitz matrices.

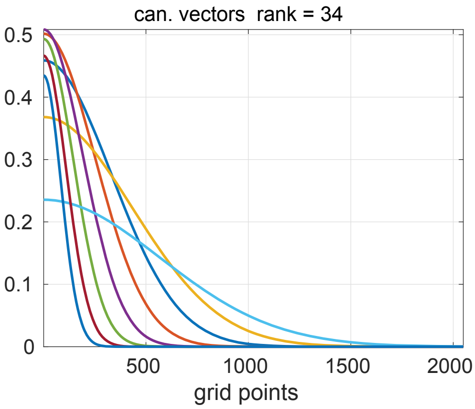

Figure 5: Selected eight canonical vectors from the full set $\mathbf{q}_k^{(1)}$, $k = 1, \dots, R$, see (3.9).

A Toeplitz matrix can be multiplied by a vector in $O(n \log n)$ operations via complementing it to the circulant matrix. In general, the inverse of a Toeplitz matrix cannot be calculated in the closed form; unlike circulant matrices, which can be diagonalized by the Fourier transform.

In the rest of this subsection, we introduce the numerical scheme, based on certain specific properties of the skeleton matrices in the symmetric rank-R decomposition (3.10) that is capable of rank-structured calculations of the analytic matrix functions $\mathcal{F}(\mathbf{C}_R)$. We discuss the most interesting examples of the functions $\mathcal{F}_1(\mathbf{C}) = \mathbf{C}^{-1}$ and $\mathcal{F}_2(\mathbf{C}) = \mathbf{C}^{\frac{1}{2}}$, where $\mathbf{C} = \mathbf{C}_R$.

Given an symmetric positive definite matrix such that $\|\mathbf{A}\| = q < 1$, the matrix-valued function is given as the exponentially fast converging series

$$\mathcal{F}(\mathbf{A}) = \mathbf{E} + a_1\mathbf{A} + a_2\mathbf{A}^2 + \cdots, \tag{3.11}$$

where the matrix $\mathbf{A}$, acting in the multidimensional index set, allows the low-rank Kronecker tensor decomposition. Then the low-rank tensor approximation of $\mathcal{F}(\mathbf{A})$ can be computed by the "add-and-compress" scheme, where each term in the series above (3.11) is summed using the rank-truncation algorithm in the corresponding format.

To limit the rank-structured evaluation of $\mathcal{F}_1(\mathbf{C})$ to the described framework, we propose the special rank-structured additive splitting of the covariance matrix $\mathbf{C}$ with the easily invertible dominating part. To that end, we construct the diagonal matrix $\mathbf{Q}_0^{(1)}$ by assembling all of the diagonal sub-matrices in $\mathbf{Q}_k^{(1)}$, $k = 1, \dots, R$ (in the following, we simplify the notation by omitting the upper index (1)):

$$\mathbf{Q}_0 := \sum_{k=1}^{R} \operatorname{diag}(\mathbf{Q}_k),$$

and modify each Toeplitz matrix $\mathbf{Q}_k$ by subtracting its diagonal part,

$$\mathbf{Q}_k \mapsto \widehat{\mathbf{Q}}_k := \mathbf{Q}_k - \operatorname{diag}(\mathbf{Q}_k), \qquad k = 1, \dots, R.$$

Using the matrices defined above, we introduce the rank $R + 1$ additive splitting of $\mathbf{C}$, which is defined by the skeleton matrices $\mathbf{Q}_0$ and $\widehat{\mathbf{Q}}_k$ since $\mathbf{C}$ is Kronecker symmetrical,

$$\mathbf{C} = \mathbf{Q}_0 \otimes \mathbf{Q}_0 \otimes \mathbf{Q}_0 + \sum_{k=1}^{R} \widehat{\mathbf{Q}}_k^{(1)} \otimes \widehat{\mathbf{Q}}_k^{(2)} \otimes \widehat{\mathbf{Q}}_k^{(3)}.$$

Hence, we have

$$\mathbf{C}^{-1} = \mathbf{Q}_0^{-1} \otimes \mathbf{Q}_0^{-1} \otimes \mathbf{Q}_0^{-1} \left(\mathbf{E} + \sum_{k=1}^{R} \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(1)} \otimes \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(2)} \otimes \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(3)} \right)^{-1}.$$

Likewise, since $\mathbf{Q}_0 \otimes \mathbf{Q}_0 \otimes \mathbf{Q}_0$ is a scaled identity, we obtain

$$\mathbf{C}^{\frac{1}{2}} = \mathbf{Q}_0^{\frac{1}{2}} \otimes \mathbf{Q}_0^{\frac{1}{2}} \otimes \mathbf{Q}_0^{\frac{1}{2}} \left(\mathbf{E} + \sum_{k=1}^{R} \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(1)} \otimes \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(2)} \otimes \mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(3)} \right)^{\frac{1}{2}}. \tag{3.12}$$

We assume that $\|\mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(1)}\| < 1$ for $k = 1, \dots, R$ in some norm; thus, we can apply the “add-and-compress” scheme described above.

We illustrate the “add-and-compress” computational scheme in the following example. We consider the covariance matrix $\mathbf{C}_R$, obtained by a rank-R sinc approximation of the Slater function $e^{-\|x\|^p}$ with $R = 40$ on a grid with $n = 1{,}025$ sampling points. Figure 6 demonstrates the decay in both the matrix norms $\mathbf{Q}_k^{(1)}$ (left), and the scaled, preconditioned matrices $\mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(1)}$, $k = 1, \dots, R$ (right). We use the scaling factor of $\frac{1}{n}$. The right figure indicates that the analytic matrix functions $\mathcal{F}(\mathbf{C}_R)$ can be evaluated by using an exponentially fast convergent power series supported by the “add-and-compress” strategy to control the tensor rank.

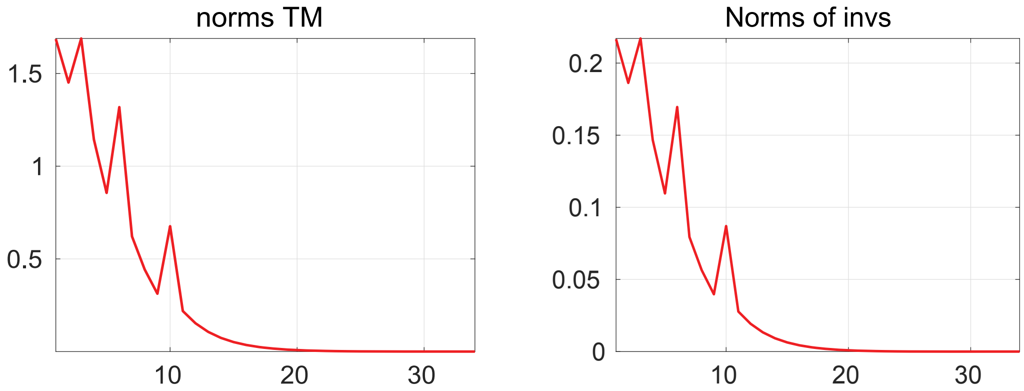

Figure 6: Scaled norms $\|\mathbf{Q}_k^{(1)}\|$ (left) and $\|\mathbf{Q}_0^{-1}\widehat{\mathbf{Q}}_k^{(1)}\|$ (right) vs. $k = 1, \dots, R$.

In the Kriging calculations (Task 3 above), the low-rank tensor structure in the covariance matrix $\mathbf{C}_R$ can be directly utilized if the sampling points in the Kriging algorithm form a smaller $m_1 \times m_2 \times m_3$ tensor subgrid of the initial $n \times n \times n$ tensor grid Ω_n with $m_\ell < n$. The same argument also applies to the evaluation of conditional covariance. In the general case of “non-tensor” locations of the sampling points, some mixed tensor factorizations could be applied.

3.7 Numerical Illustrations

In what follows, we check some examples of the low-rank Tucker tensor approximation of the p-Slater function $C(x) = e^{-\|x\|^p}$, and Matérn kernels with full-grid tensor representation. We demonstrate the fast exponential convergence of the tensor approximation in the Tucker rank. The functions were sampled on the $n_1 \times n_2 \times n_3$ three-dimensional Cartesian grid with $n_\ell = 100$, $\ell = 1, 2, 3$.

For a continuous function $q : \Omega \to \mathbb{R}$, where $\Omega := \prod_{\ell=1}^{d}[-b_\ell, b_\ell] \subset \mathbb{R}^d$, and $0 < b_\ell < \infty$, we introduce the collocation-type function-related tensor of order d:

$$\mathbf{Q} \equiv \mathbf{Q}(q) := [q_{i_1 \ldots i_d}] \in \mathbb{R}^{I_1 \times \cdots \times I_d} \quad \text{with } q_{i_1 \ldots i_d} := q(x_{i_1}^{(1)}, \ldots, x_{i_d}^{(d)}),$$

where $(x_{i_1}^{(1)}, \ldots, x_{i_d}^{(d)}) \in \mathbb{R}^d$ are grid collocation points, indexed by $\mathcal{I} = I_1 \times \cdots \times I_d$,

$$x_{i_\ell}^{(\ell)} = -b_\ell + (i_\ell - 1)h_\ell, \quad i_\ell = 1, 2, \ldots, n_\ell, \ \ell = 1, \ldots, d,$$

which are the nodes of equally spaced subintervals with a mesh size of $h_\ell = \frac{2b_\ell}{n_\ell - 1}$.

We test the convergence of the error in the relative Frobenius norm with respect to the Tucker rank for p-Slater functions with $p = 0.1, 0.2, 1.9, 2.0$. The Frobenius norm is computed as

$$E_{FN} = \frac{\|\mathbf{Q} - \mathbf{Q}_{(r)}\|}{\|\mathbf{Q}\|}, \tag{3.13}$$

where $\mathbf{Q}_{(r)}$ is the tensor reconstructed from the Tucker rank-r decomposition of $\mathbf{Q}$.

Figure 7 shows convergence of the Frobenius error with respect to the Tucker rank for the p-Slater function discretized on the three-dimensional Cartesian grid

$$C(x, y) = e^{-\|x-y\|^p} \tag{3.14}$$

for different values of the parameter p. These functions are illustrated in Figure 8 for $p = 0.1$ and $p = 1.9$. Figure 9 shows for a Slater function with $p = 1$ the dependence of the Tucker decomposition error in the Frobenius norm (3.13) on the Tucker rank, for the increasing grid parameter n.

Figure 10 shows the convergence with respect to the Tucker rank for the spectral density of Matérn covariance

$$f_{\alpha,\nu}(\rho) := \frac{C}{(\alpha^2 + \rho^2)^{\nu + \frac{d}{2}}}, \tag{3.15}$$

where $\alpha \in (0.1, 100)$ and $d = 1, 2, 3$. The Tucker decomposition rank is strongly dependent on the parameter α and weakly depend on the parameter ν. The three-dimensional Matérn functions with the parameters $\nu = 0.4$, $\alpha = 0.1$ (left) and $\nu = 0.4$, $\alpha = 100$ (right) are presented in Figure 11, showing the function at $z = 0$.

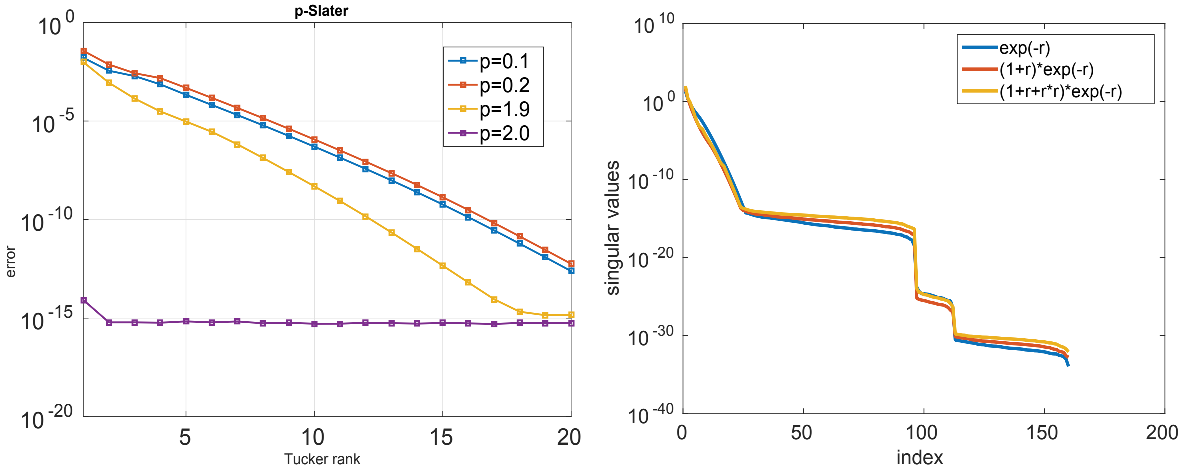

Figure 7: Convergence in the Frobenius error (3.13) with respect to the Tucker rank for the function (3.14) with $p = 0.1, 0.2, 1.9, 2.0$ (left); Decay of singular values of the weighted Slater function (right).

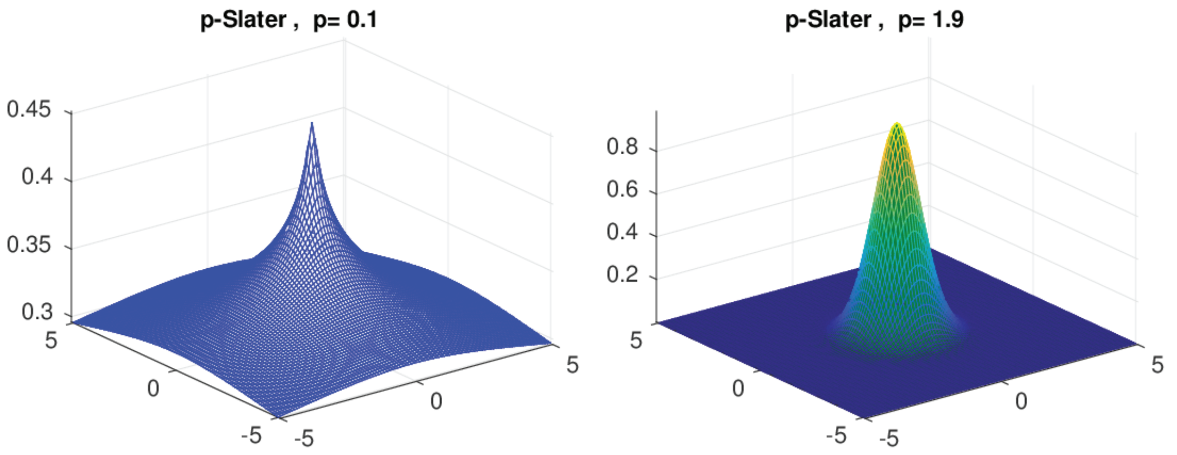

Figure 8: Cross section of the three-dimensional radial function (3.14) with $p = 0.1$ (left) and $p = 1.9$ (right) at level $z = 0$.

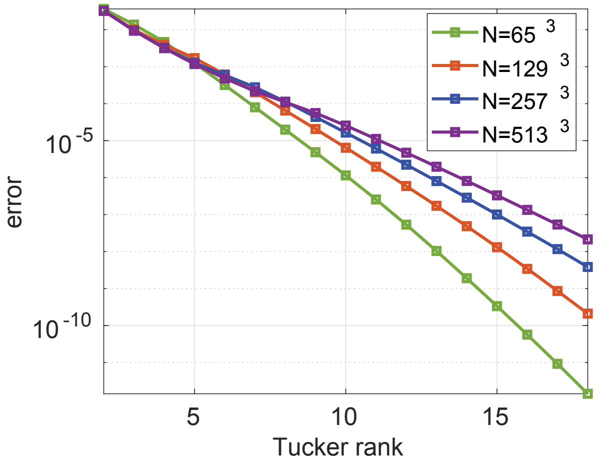

Figure 9: Multigrid Tucker: convergence with respect to Tucker ranks of a Slater function with $p = 1$ on a sequence of grids.

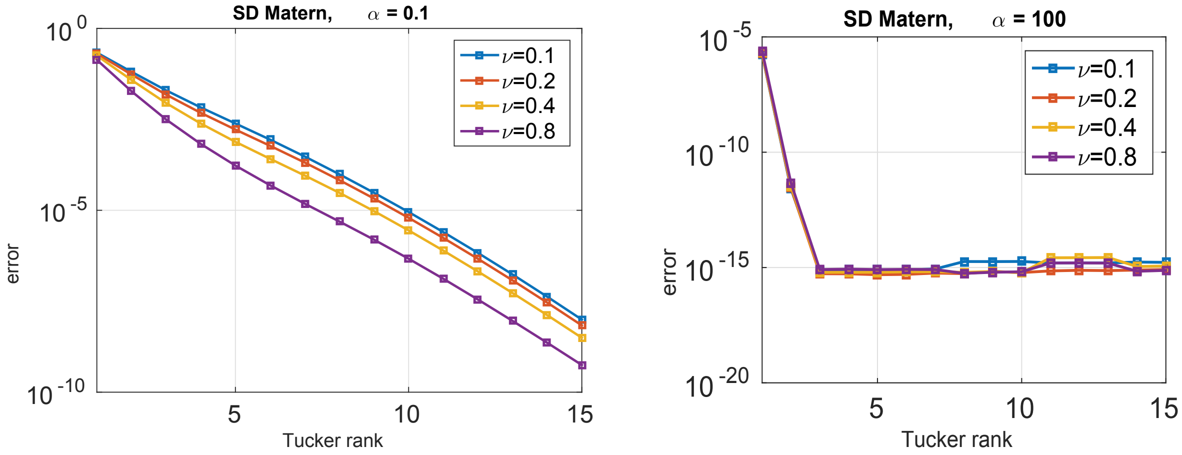

Figure 10: Convergence with respect to the Tucker rank of three-dimensional spectral density of Matérn covariance (3.15) with $\alpha = 0.1$ (left) and $\alpha = 100$ (right).

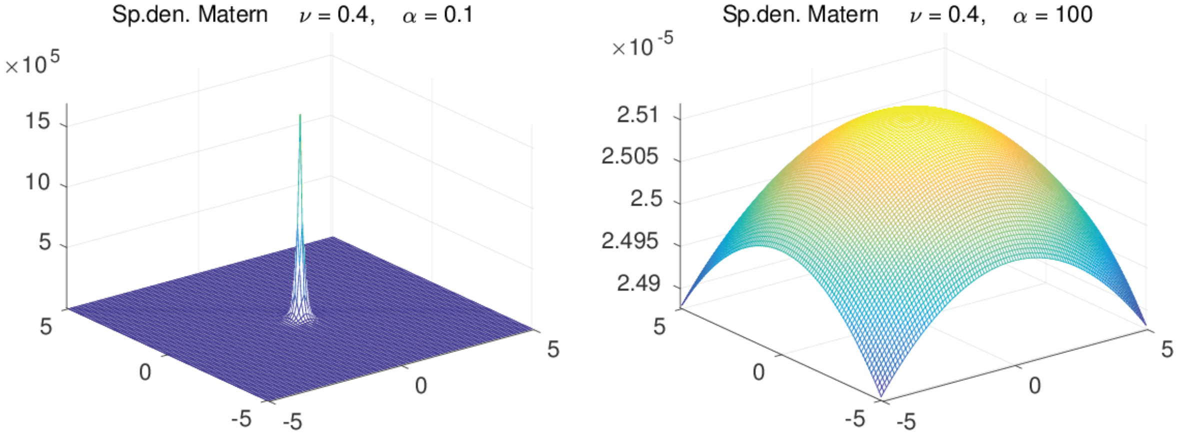

Figure 11: The shape of three-dimensional spectral density of Matérn covariance (3.15) with $\alpha = 0.1$ (left) and $\alpha = 100$ (right).

These numerical experiments demonstrate the good algebraic separability of the typical multidimensional functions used in spatial statistics, that lead us to apply the low-rank tensor decomposition methods to the multidimensional problems of statistical data analysis.

4 Solutions to Typical Tasks in Low-Rank Tensor Format

In this section, we walk through the solutions to the statistical questions raised in the motivation above, Section 2. We add some lemmas to summarize the new computing and storage costs. Let $N = n^d$, measurement vector $\mathbf{z} \in \mathbb{R}^m$, $\mathbf{C}_{ss} \in \mathbb{R}^{N\times N}$, $\mathbf{C}_{zz} \in \mathbb{R}^{m\times m}$, and $\mathbf{C}_{sz} \in \mathbb{R}^{N\times m}$. We also introduce the restriction operator P, which consists of only ones and zeros, and pick sub-indices $\{i_1, \dots, i_m\}$ from the whole index set $\{i_1, \dots, i_N\}$. This operator P has tensor-1 structure, i.e., $P = \bigotimes_{\nu=1}^{d} P_\nu$. An application of this restriction tensor does not change tensor ranks.

Computing Matrix-Vector Product. Let $\mathbf{C} = \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}$. If $\mathbf{z}$ is separable, i.e., $\|\mathbf{z} - \sum_{j=1}^{r_b} \bigotimes_{\nu=1}^{d} \mathbf{z}_{j\nu}\| \le \varepsilon$, then

$$\mathbf{Cz} = \sum_{i=1}^{r} \sum_{j=1}^{r_z} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}\mathbf{z}_{j\mu}.$$

If $\mathbf{z}$ is non-separable, then low-rank tensor properties cannot be employed and either the FFT idea [51] or the hierarchical matrix technique should be applied instead [19, 21, 22, 38].

Lemma 4.1. *The computing cost of the product* $\mathbf{Cz}$ *is reduced from* $\mathcal{O}(N^2)$ *to* $\mathcal{O}(rr_z dn^2)$, *where* $N = n^d$, $d \ge 1$.

Trace and Diagonal of C. Let $\mathbf{C} \approx \tilde{\mathbf{C}} = \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}$. Then

$$\operatorname{diag}(\tilde{\mathbf{C}}) = \operatorname{diag}\left(\sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}\right) = \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \operatorname{diag}(\mathbf{C}_{i\mu}), \tag{4.1}$$

$$\operatorname{trace}(\tilde{\mathbf{C}}) = \operatorname{trace}\left(\sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}\right) = \sum_{i=1}^{r} \prod_{\mu=1}^{d} \operatorname{trace}(\mathbf{C}_{i\mu}). \tag{4.2}$$

The proof follows from the properties of the Kronecker tensors.

Lemma 4.2. *The cost of computing the right-hand sides in* (4.1)–(4.2) *is* rdn, *where* $\mathbf{C}_{i\mu} \in \mathbb{R}^{n\times n}$.

For simplicity, we assume that $n_1 = n_2 = \cdots = n_d = n$ and $\sum_{i=1}^{d} n_i = dn$.

Lemma 4.3. *The computing cost of* $\operatorname{diag}(\mathbf{C})$ *and* $\operatorname{trace}(\mathbf{C})$ *is reduced from* $\mathcal{O}(N)$ *to* $\mathcal{O}(rdn)$. *The cost of* $\det(\mathbf{C})$ *is reduced from* $\mathcal{O}(N^3)$ *to* $\mathcal{O}(dn^3)$.

	n		
d	**100**	**500**	**1000**
1000	3.7	67	491

Table 1: Computing time (in seconds) to set up and compute the trace of $\tilde{\mathbf{C}} = \sum_{j=1}^{r} \bigotimes_{\nu=1}^{d} \mathbf{C}_{j\nu}$, $r = 10$, $\tilde{\mathbf{C}} \in \mathbb{R}^{N\times N}$, where $N = n^d$, $d = 1{,}000$ and $n = \{100, 500, 1000\}$. A modern desktop computer with 128 GB RAM was used.

	Tucker rank *r*									
n	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
129	0.386	0.20	0.12	0.07	0.04	0.017	0.002	1.2e−4	8.4e−6	7.5e−6
257	0.386	0.20	0.12	0.073	0.046	0.029	0.017	0.007	8.0e−4	1.4e−5
513	0.386	0.20	0.12	0.073	0.047	0.031	0.020	0.0138	0.008	0.0035

Table 2: The error of the Tucker approximation to the value **A** at the origin (the exact value is equal to 1) versus the Tucker rank and the grid size $n = 2k + 1$.

Example 4.1. A simple Matlab test for computing trace($\mathbf{C}$) on a working station with 128 GB produces the computing times in seconds shown in Table 1.

In what follows, we discuss the computation of trace and diag in the case of Tucker representation of the generating Matérn function on $(2k+1)^{\otimes d}$ grid and present the corresponding numerical example for $d = 3$. We notice that due to Toeplitz structure of the skeleton matrices of size $(2k+1) \times (2k+1)$, the diagonal of the covariance matrix $\mathbf{C}$ is the weighted Tucker sum of the Kronecker products of scaled identity matrices in $\mathbb{R}^{2k+1}$. The scaling factor is determined by the value of generating Tucker vector $\mathbf{v}_{\nu_\ell}^{(\ell)}(k+1)$ corresponding the origin of the computational box $[-b, b]^d$. Let the matrix $\mathbf{C}$ be composed by using the Tucker tensor $\mathbf{A}$ approximating the generating Matérn function. Then, as a straightforward consequence of the above remark, we derive the simple representations

$$\operatorname{diag}(\mathbf{C}) = \mathbf{A}(x_0)\mathbf{E}^{(d)}, \quad \operatorname{trace}(\mathbf{C}) = \mathbf{A}(x_0)(2k+1)^d,$$

where x_0 corresponds to the origin $x = 0$ in $[-b, b]^d$ and $\mathbf{E}^{(d)}$ is the identity matrix in the full tensor space. The grid coordinate of x_0 is determined by the multi-index $(k+1, \dots, k+1)$.

Table 2 represents the values of $\mathbf{A}(x_0)$ computed by the Tucker approximation to the three-dimensional Slater function $e^{-\|x\|}$ on the $(2k+1)^{\otimes 3}$ grids with $n = 2k + 1 = 129, 257, 513$, and for different Tucker rank parameters $r = 1, 2, \dots, 10$.

Given the rank-$\mathbf{r}$ Tucker tensor $\mathbf{A}$, the complexity for calculation of $\mathbf{A}(x_0)$ is estimated by $O(r^d)$.

Computing Square Root $\mathbf{C}^{\frac{1}{2}}$. Observe that $\mathbf{C}^{\frac{1}{2}}$ can be computed as in (3.12). An iterative method for computing $\mathbf{C}^{\frac{1}{2}}$ is presented in [16].

Linear Solvers in a Low-Rank Tensor Format. Likely, there is already a good theory for solving linear systems $\mathbf{C}\mathbf{w} = \mathbf{z}$ with symmetric and positive definite matrix $\mathbf{C}$, in a tensor format. We refer to the overview works [18, 20, 34, 35]. Some particular linear solvers are developed in [2, 12, 13, 24, 36], [11, 36]. We also recommend to use the Tensor Toolbox [41], which contains routines for CP and Tucker tensor formats.

4.1 Computing $\mathbf{z}^T\mathbf{C}^{-1}\mathbf{z}$

Lemma 4.4. *Let $\|\mathbf{z} - \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{z}_{i\mu}\| \le \varepsilon$. We assume that there is an iterative method that can be used to solve the linear system $\mathbf{C}\mathbf{w} = \mathbf{z}$ in a low-rank tensor format and to find the solution in the form $\mathbf{w} = \sum_{i=1}^{r} \bigotimes_{\mu=1}^{d} \mathbf{w}_{i\mu}$. Then the quadratic form $\mathbf{z}^T\mathbf{C}^{-1}\mathbf{z}$ is the following scalar products:*

$$\mathbf{z}^T\mathbf{C}^{-1}\mathbf{z} = \sum_{i=1}^{r} \sum_{j=1}^{r_z} \prod_{\mu=1}^{d} (\mathbf{w}_{i\mu}, \mathbf{z}_{j\mu}).$$

The proof follows from the definition and properties of the tensor and scalar products. If $\mathbf{z}$ is non-separable, then low-rank tensor properties cannot be employed and the FFT idea [51] or the hierarchical matrix technique [38] should be employed.

Lemma 4.5. *The computing cost of the quadratic form $\mathbf{z}^T\mathbf{C}^{-1}\mathbf{z}$ is the product of the number of required iterations and the cost of one iteration, which is $\mathcal{O}(rr_z dm^2)$ (assuming that the iterative method required only matrix-vector products).*

The proof follows from the definitions and properties of the tensor and scalar products.

4.2 Interpolation by Simple Kriging

The three most computationally demanding tasks in Kriging are:
(1) solving an $M \times M$ system of equations to obtain the Kriging weights,
(2) obtaining the $N \times 1$ Kriging estimate by superposing the Kriging weights with the $N \times M$ cross-covariance matrix between the measurements and the unknowns,
(3) evaluating the $N \times 1$ estimation variance as the diagonal of an $N \times N$ conditional covariance matrix [51].
Here, M refers to the number of measured data values, and N refers to the number of estimation points. When optimizing the design of the sampling patterns, the challenge is to evaluate the scalar measures of the $N \times N$ conditional covariance matrix (see φ_A and φ_C in equation (2.1) and Task 4) repeatedly within a high-dimensional and non-linear optimization procedure (e.g., [42, 58]).

The following Kriging formula is well known [51]:

$$\hat{\mathbf{s}} = \mathbf{C}_{sz}\mathbf{C}_{zz}^{-1}\mathbf{z}.$$

Lemma 4.6. *If $\|\mathbf{C}_{sz} - \sum_{i=1}^{r_C} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}\| \le \varepsilon$ for some small $\varepsilon \ge 0$ and Lemma 4.4 holds, then*

$$\mathbf{C}_{sz}\mathbf{C}_{zz}^{-1}\mathbf{z} \approx \sum_{i=1}^{r_z}\sum_{j=1}^{r_C}\bigotimes_{\nu=1}^{d} \mathbf{C}_{i\nu}\mathbf{w}_{j\nu}. \tag{4.3}$$

The proof follows from the definitions and the properties of the tensor and scalar products.

Lemma 4.7. *The computing cost of solving the linear system $\mathbf{C}_{zz}^{-1}\mathbf{z}$ is $\mathcal{O}(\#\text{iters} \cdot r_z r d m^2)$. Computation of the Kriging coefficients by equation (4.3) costs $\mathcal{O}(r_z r_C dnm) + \mathcal{O}(\#\text{iters} \cdot r_z r d m^2)$.*

If $\mathbf{z}$ is non-separable, then the low-rank tensor properties cannot be employed and either the FFT idea [51] or the hierarchical matrix technique [38] should be applied.

4.3 Computing Conditional Covariance

Let $\mathbf{y} \in \mathbb{R}^m$ be the vector of measurements. The conditional covariance matrix is

$$\mathbf{C}_{ss|y} = \mathbf{C}_{ss} - \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys}.$$

The associated estimation variance $\hat{\boldsymbol{\sigma}}$ is the diagonal of the $N \times N$ conditional covariance matrix $\mathbf{C}_{ss|y}$:

$$\hat{\boldsymbol{\sigma}}_{\mathbf{s}} = \operatorname{diag}(\mathbf{C}_{ss|y}) = \operatorname{diag}(\mathbf{C}_{ss} - \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys}). \tag{4.4}$$

Let us assume that the measurements are taken at locations that form a subset of the total set of nodes $\mathcal{I} = \{0, \dots, N-1\}$, i.e., $\mathcal{I}_{\mathcal{M}} = \{i_1, \dots, i_m\} \subset \mathcal{I}$. We also assume that the nodes $\mathcal{I}_{\mathcal{M}}$ belong to a tensor mesh, i.e., if $\mathcal{I} = \bigotimes_{\nu=1}^{d} I_\nu$ and $\mathcal{I}_{\mathcal{M}} = \bigotimes_{\nu=1}^{d} \hat{I}_\nu$, then $\hat{I}_\nu \subseteq I_{nu}$.

Let

$$\mathbf{C}_{yy} = \sum_{k=1}^{r}\bigotimes_{\mu=1}^{d} \mathbf{C}_{k\mu}.$$

Again, we use low-rank tensor solvers, this time to solve the matrix system $\mathbf{C}_{yy}\mathbf{W} = \mathbf{C}_{ys}$. We obtain the solution

$$\mathbf{W} = \mathbf{C}_{yy}^{-1}\mathbf{C}_{ys} = \sum_{j=1}^{r_w} \bigotimes_{\mu=1}^{d} \mathbf{C}_{j\mu}.$$

Assuming that $\mathbf{C}_{sy} \approx \sum_{i=1}^{r_C} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}$, we obtain

$$\mathbf{C}_{sy}\mathbf{W} = \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys} = \sum_{i=1}^{r_C} \bigotimes_{\nu=1}^{d} \mathbf{C}_{i\nu} \sum_{j=1}^{r_w} \bigotimes_{\mu=1}^{d} \mathbf{C}_{j\mu} = \sum_{i=1}^{r_C} \sum_{j=1}^{r_w} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}\mathbf{C}_{j\mu} \approx \sum_{j=1}^{r_0} \bigotimes_{\mu=1}^{d} \tilde{\mathbf{C}}_{j\mu},$$

where $r_0 > 0$ is the new rank after a rank-truncation procedure and $\tilde{\mathbf{C}}_{j\mu}$ are new factors. Let

$$\mathbf{C}_{ss} = \sum_{i=1}^{r_s} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu}.$$

The conditional covariance is

$$\mathbf{C}_{ss|y} = \sum_{i=1}^{r_s} \bigotimes_{\mu=1}^{d} \mathbf{C}_{i\mu} - \sum_{i=1}^{r_0} \bigotimes_{\mu=1}^{d} \tilde{\mathbf{C}}_{i\mu} = \sum_{i=1}^{r_s+r_0} \bigotimes_{\mu=1}^{d} \hat{\mathbf{C}}_{i\mu},$$

where $\hat{\mathbf{C}}_{i\mu} = \mathbf{C}_{i\mu}$ for $1 \le i \le r_s$ and $\hat{\mathbf{C}}_{i\mu} = -\tilde{\mathbf{C}}_{i\nu}$ for $r_s < i \le r_0 + r_s$.

4.4 Example: Separable Covariance Matrices

Let

$$\operatorname{cov}(\mathbf{x}, \mathbf{y}) = \exp^{-|\mathbf{x}-\mathbf{y}|^2}$$

be the Gaussian covariance function, where $\mathbf{x} = (x_1, \dots, x_d)$ and $\mathbf{y} = (y_1, \dots, y_d) \in \mathcal{D} \subset \mathbb{R}^d$. The function $\operatorname{cov}(\mathbf{x}, \mathbf{y})$ can be written as a tensor product of one-dimensional functions:

$$\operatorname{cov}(\mathbf{x}, \mathbf{y}) = \exp^{-|x_1-y_1|^2} \otimes \cdots \otimes \exp^{-|x_d-y_d|^2}.$$

After discretization of $\operatorname{cov}(\mathbf{x}, \mathbf{y})$, we obtain $\mathbf{C}$ as a rank-1 Kronecker product of the one-dimensional covariance matrices, i.e.,

$$\mathbf{C} = \mathbf{C}_1 \otimes \cdots \otimes \mathbf{C}_d. \tag{4.5}$$

We note that arbitrary discretization (anisotropy) can occur in any direction.

Lemma 4.8. *If d Cholesky decompositions exist, i.e.,* $\mathbf{C}_i = \mathbf{L}_i \cdot \mathbf{L}_i^T$ *and* $i = 1, \dots, d$, *then*

$$\mathbf{C}_1 \otimes \cdots \otimes \mathbf{C}_d = (\mathbf{L}_1\mathbf{L}_1^T) \otimes \cdots \otimes (\mathbf{L}_d\mathbf{L}_d^T) = (\mathbf{L}_1 \otimes \cdots \otimes \mathbf{L}_d) \cdot (\mathbf{L}_1^T \otimes \cdots \otimes \mathbf{L}_d^T) =: \mathbf{L} \cdot \mathbf{L}^T,$$

where $\mathbf{L} := \mathbf{L}_1 \otimes \cdots \otimes \mathbf{L}_d$ *and* $\mathbf{L}^T := \mathbf{L}_1^T \otimes \cdots \otimes \mathbf{L}_d^T$ *are also lower- and upper-triangular matrices, respectively.*

Lemma 4.8 shows that

(a) the Gaussian covariance function in dimensions $d > 1$ can be written as the tensor sum of one-dimensional covariance functions,

(b) its Cholesky factor can be computed via Cholesky factors computed from one-dimensional covariances.

The computational complexity drops from $\mathcal{O}(N \log N)$, $N = n^d$, to $\mathcal{O}(dn \log n)$, where n is the number of mesh points in a one-dimensional problem. Further research is required on non-Gaussian covariance functions.

Lemma 4.9. *Let* $\mathbf{C} = \mathbf{C}_1 \otimes \cdots \otimes \mathbf{C}_d$. *If the inverse matrices* $\mathbf{C}_i^{-1}$, $i = 1, \dots, d$, *exist, then*

$$(\mathbf{C}_1 \otimes \cdots \otimes \mathbf{C}_d)^{-1} = \mathbf{C}_1^{-1} \otimes \cdots \otimes \mathbf{C}_d^{-1}.$$

The computational complexity drops from $\mathcal{O}(N \log N)$, $N = n^d$, to $\mathcal{O}(dn \log n)$, where n is the number of mesh points in a one-dimensional problem.

Remark 4.10. We assume here that we have an efficient method to invert $\mathbf{C}$ (e.g., FFT or hierarchical matrices) with a cost of $\mathcal{O}(N\log N)$. If not, then the complexity cost drops from $\mathcal{O}(N^3)$ to $\mathcal{O}(dn^3)$ (usual Gaussian elimination algorithm).

Lemma 4.11. *If* $\mathbf{C}_i$, $i = 1,\dots,d$, *are covariance matrices, then we can compute* $\log\det\mathbf{C}$, *where* $\mathbf{C}$ *is a separable rank-*1 *d-dimensional covariance function, as*

$$\log\det(\mathbf{C}_1\otimes\cdots\otimes\mathbf{C}_d) = \sum_{j=1}^{d}\log\det\mathbf{C}_j\prod_{i=1,i\neq j}^{d} n_i. \tag{4.6}$$

Proof. We check for $d = 2$ that

$$\det(\mathbf{C}_1\otimes\mathbf{C}_2) = \det(\mathbf{C}_1)^{n_2}\cdot\det(\mathbf{C}_2)^{n_1}$$

and then apply mathematical induction. □

The computational cost drops again from $\mathcal{O}(N\log N)$, $N = n^d$, to $\mathcal{O}(dn\log n)$. A similar assumption to Remark 4.10 for computing $\det(\mathbf{C})$ also holds here.

Example 4.2. Let $n = 6000$, $d = 3$, and $N = 6000^3$. Using MATLAB on a MacBookPro with 16 GB RAM, the time required set up the matrices $\mathbf{C}_1$,$\mathbf{C}_2$, and $\mathbf{C}_3$ is 11 seconds; it takes 4 seconds to compute $\mathbf{L}_1$, $\mathbf{L}_2$, and $\mathbf{L}_3$. The large matrices $\mathbf{C}$ and $\mathbf{L}$ are never constructed (i.e., the Kronecker product is never calculated).

Example 4.3. In previous work [43] we used the hierarchical matrix technique to approximate $\mathbf{C}_i$ and its Cholesky factor $\mathbf{L}_i$ for $n = 2\cdot10^6$ in 2 minutes. Here, we combine the hierarchical matrix technique and the Kronecker tensor product. Assuming $\mathbf{C} = \mathbf{C}_1\otimes\cdots\otimes\mathbf{C}_d$, we approximate $\mathbf{C}$ for $n = (2\cdot10^6)^d$ in $2d$ minutes.

Lemma 4.12. *Let* $\mathbf{C}$ *be the same as in equation* (4.5), *and let Lemma 4.4 hold. Then we apply the property in equation* (4.6) *to obtain a tensor approximation of the log-likelihood:*

$$\mathcal{L}\approx\tilde{\mathcal{L}} = -\frac{\prod_{\nu=1}^{d} n_\nu}{\log(2\pi)} - \sum_{j=1}^{d}\log\det\mathbf{C}_j\prod_{i=1,i\neq j}^{d} n_i - \sum_{i=1}^{r}\sum_{j=1}^{r}\prod_{\nu=1}^{d}(\mathbf{u}_{i,\nu}^T,\mathbf{u}_{j,\nu}). \tag{4.7}$$

Equation (4.7) shows one disadvantage of the Gaussian log-likelihood function in high dimensions. Namely, the log-likelihood grows exponentially with d as n^d.

5 Conclusion

In this work, we demonstrate that the basic functions and operators used in spatial statistics may be represented using rank-structured tensor formats and that the error of this representation exhibits the exponential decay with respect to the tensor rank. We applied the Tucker and canonical tensor decompositions to a family of Matérn-type and Slater-type functions with varying parameters and demonstrated numerically that their approximations exhibit exponentially fast convergence. A low-rank tensor approximation of the Matérn covariance function and its Fourier transform is considered. We separated the radial basis functions using the Laplace transforms to prove the existence of such low-rank approximations, and applied the sinc quadrature method to estimate the tensor ranks and accuracy.

We also demonstrated how to compute $\operatorname{diag}(\mathbf{C})$, $\operatorname{trace}(\mathbf{C})$, the matrix-vector product, Kriging operations, and the geostatistical optimal design in a low-rank tensor format with at a linear cost. For matrix $\mathbf{C}$ of size $N\times N$, $N = 6000^3$ and of tensor rank 1, we were able to compute the Cholesky factorization in 15 seconds. We also computed the Tucker approximation to the three-dimensional Slater function $e^{-\|x\|}$ on the grid with 513^3 points and Tucker ranks $r = 1,2,\dots,10$. Furthermore, we demonstrated how to compute $\operatorname{trace}(\mathbf{C})$ for $N = n^d = 1000^{1000}$. This might be useful in machine learning. In this paper, operations such as computing the Cholesky factorization, inverse, and determinant have been implemented for rank-1 tensors (e.g., the Gaussian covariance has a tensor rank-1). These formulas could be useful for developing successive rank-1 updates in greedy algorithms. Further investigations are needed for the representation of these quantities with the ranks higher than one.

Additionally, in Section 3.7 we studies the influence of the parameters of the Matérn covariance function on the tensor ranks (Figures 7 and 9). We observed (see Figure 10) that the dependence of the parameters of the Matérn covariance function on the tensor ranks is very weak, and the ranks grew slowly. In this paper, we also highlighted that big data statistical problems can be effectively treated by using the special low-rank tensor techniques.

Funding: The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST).

References

[1] S. Ambikasaran, J. Y. Li, P. K. Kitanidis and E. Darve, Large-scale stochastic linear inversion using hierarchical matrices, *Comput. Geosci.* **17** (2013), no. 6, 913–927.

[2] J. Ballani and D. Kressner, Sparse inverse covariance estimation with hierarchical matrices, preprint (2015), http://sma.epfl.ch/~anchpcommon/publications/quic_ballani_kressner_2014.pdf.

[3] C. Bertoglio and B. N. Khoromskij, Low-rank quadrature-based tensor approximation of the Galerkin projected Newton/Yukawa kernels, *Comput. Phys. Commun.* **183** (2012), no. 4, 904–912.

[4] S. Börm and J. Garcke, Approximating gaussian processes with H^2-matrices, in: *Proceedings of 18th European Conference on Machine Learning—ECML 2007*, Lecture Notes in Artificial Intelligence 4701, Springer, Berlin (2007), 42–53.

[5] S. F. Boys, G. B. Cook, C. M. Reeves and I. Shavitt, Automatic fundamental calculations of molecular structure, *Nature* **178** (1956), 1207–1209.

[6] D. Braess, *Nonlinear Approximation Theory*, Springer Ser. Comput. Math. 7, Springer, Berlin, 1986.

[7] J.-P. Chilès and P. Delfiner, *Geostatistics*, Wiley Ser. Probab. Stat., John Wiley & Sons, New York, 1999.

[8] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*, Wiley, New York, 2002.

[9] S. De Iaco, S. Maggio, M. Palma and D. Posa, Toward an automatic procedure for modeling multivariate space-time data, *Comput. Geosci.* **41** (2011), DOI 10.1016/j.cageo.2011.08.008.

[10] L. De Lathauwer, B. De Moor and J. Vandewalle, A multilinear singular value decomposition, *SIAM J. Matrix Anal. Appl.* **21** (2000), no. 4, 1253–1278.

[11] S. Dolgov, B. N. Khoromskij, A. Litvinenko and H. G. Matthies, Computation of the response surface in the tensor train data format, preprint (2014), https://arxiv.org/abs/1406.2816.

[12] S. Dolgov, B. N. Khoromskij, A. Litvinenko and H. G. Matthies, Polynomial chaos expansion of random coefficients and the solution of stochastic partial differential equations in the tensor train format, *SIAM/ASA J. Uncertain. Quantif.* **3** (2015), no. 1, 1109–1135.

[13] S. Dolgov, B. N. Khoromskij and D. Savostyanov, Superfast Fourier transform using QTT approximation, *J. Fourier Anal. Appl.* **18** (2012), no. 5, 915–953.

[14] P. A. Finke, D. J. Brus, M. F. P. Bierkens, T. Hoogland, M. Knotters and F. De Vries, Mapping groundwater dynamics using multiple sources of exhaustive high resolution data, *Geoderma* **123** (2004), no. 1, 23–39.

[15] R. Furrer and M. G. Genton, Aggregation-cokriging for highly multivariate spatial data, *Biometrika* **98** (2011), no. 3, 615–631.

[16] I. P. Gavrilyuk, W. Hackbusch and B. N. Khoromskij, Data-sparse approximation to a class of operator-valued functions, *Math. Comp.* **74** (2005), no. 250, 681–708.

[17] I. P. Gavrilyuk, W. Hackbusch and B. N. Khoromskij, Hierarchical tensor-product approximation to the inverse and related operators for high-dimensional elliptic problems, *Computing* **74** (2005), no. 2, 131–157.

[18] L. Grasedyck, D. Kressner and C. Tobler, A literature survey of low-rank tensor approximation techniques, *GAMM-Mitt.* **36** (2013), no. 1, 53–78.

[19] W. Hackbusch, A sparse matrix arithmetic based on $\mathscr{H}$-matrices. I. Introduction to $\mathscr{H}$-matrices, *Computing* **62** (1999), no. 2, 89–108.

[20] W. Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, Springer Ser. Comput. Math. 42, Springer, Heidelberg, 2012.

[21] W. Hackbusch, *Hierarchical Matrices: Algorithms and Analysis*, Springer Ser. Comput. Math. 49, Springer, Heidelberg, 2015.

[22] W. Hackbusch and B. N. Khoromskij, A sparse $\mathscr{H}$-matrix arithmetic. II. Application to multi-dimensional problems, *Computing* **64** (2000), no. 1, 21–47.

[23] W. Hackbusch and B. N. Khoromskij, Low-rank Kronecker-product approximation to multi-dimensional nonlocal operators. I. Separable approximation of multi-variate functions, *Computing* **76** (2006), no. 3–4, 177–202.

[24] W. Hackbusch and B. N. Khoromskij, Low-rank Kronecker-product approximation to multi-dimensional nonlocal operators. II. HKT representation of certain operators, *Computing* **76** (2006), no. 3–4, 203–225.

[25] M. S. Handcock and M. L. Stein, A Bayesian analysis of Kriging, *Technometrics* **35** (1993), 403–410.

[26] H. Harbrecht, M. Peters and M. Siebenmorgen, Efficient approximation of random fields for numerical applications, *Numer. Linear Algebra Appl.* **22** (2015), no. 4, 596–617.

[27] J. Håstad, Tensor rank is NP-complete, *J. Algorithms* **11** (1990), no. 4, 644–654.

[28] M. R. Haylock, N. Hofstra, A. M. Klein Tank, E. J. Klok, P. D. Jones and M. New, A european daily high-resolution gridded data set of surface temperature and precipitation for 1950–2006, *J. Geophys. Res.* **113** (2008), DOI 10.1029/2008JD010201.

[29] F. L. Hitchcock, The expression of a tensor or a polyadic as a sum of products, *J. Math. Phys.* **6** (1927), 164–189.

[30] A. G. Journel and C. J. Huijbregts, *Mining Geostatistics*, Academic Press, New York, 1978.

[31] V. Khoromskaia, Computation of the Hartree–Fock exchange by the tensor-structured methods, *Comput. Methods Appl. Math.* **10** (2010), no. 2, 204–218.

[32] V. Khoromskaia and B. N. Khoromskij, Fast tensor method for summation of long-range potentials on 3D lattices with defects, *Numer. Linear Algebra Appl.* **23** (2016), no. 2, 249–271.

[33] B. N. Khoromskij, Structured rank-$(R_1, \ldots, R_D)$ decomposition of function-related tensors in $\mathbb{R}^D$, *Comput. Methods Appl. Math.* **6** (2006), no. 2, 194–220.

[34] B. N. Khoromskij, Tensors-structured numerical methods in scientific computing: Survey on recent advances, *Chemometr. Intell. Laboratory Syst.* **110** (2011), no. 1, 1–19.

[35] B. N. Khoromskij, Tensor numerical methods for multidimensional PDEs: Theoretical analysis and initial applications, in: *CEMRACS 2013—Modelling and Simulation of Complex Systems: Stochastic and Deterministic Approaches*, ESAIM Proc. Surveys 48, EDP Sci., Les Ulis (2015), 1–28.

[36] B. N. Khoromskij and V. Khoromskaia, Low rank Tucker-type tensor approximation to classical potentials, *Cent. Eur. J. Math.* **5** (2007), no. 3, 523–550.

[37] B. N. Khoromskij and V. Khoromskaia, Multigrid accelerated tensor approximation of function related multidimensional arrays, *SIAM J. Sci. Comput.* **31** (2009), no. 4, 3002–3026.

[38] B. N. Khoromskij, A. Litvinenko and H. G. Matthies, Application of hierarchical matrices for computing the Karhunen–Loève expansion, *Computing* **84** (2009), no. 1–2, 49–67.

[39] P. K. Kitanidis, *Introduction to Geostatistics*, Cambridge University Press, Cambridge, 1997.

[40] T. G. Kolda, Orthogonal tensor decompositions, *SIAM J. Matrix Anal. Appl.* **23** (2001), no. 1, 243–255.

[41] T. G. Kolda and B. W. Bader, Tensor decompositions and applications, *SIAM Rev.* **51** (2009), no. 3, 455–500.

[42] J. B. Kollat, P. M. Reed and J. R. Kasprzyk, A new epsilon-dominance hierarchical bayesian optimization algorithm for large multiobjective monitoring network design problems, *Adv. Water Res.* **31** (2008), no. 5, 828–845.

[43] A. Litvinenko, HLIBCov: Parallel hierarchical matrix approximation of large covariance matrices and likelihoods with applications in parameter identification, preprint (2017), https://arxiv.org/abs/1709.08625.

[44] A. Litvinenko, Y. Sun, M. G. Genton and D. Keyes, Likelihood approximation with hierarchical matrices for large spatial datasets, preprint (2017), https://arxiv.org/abs/1709.04419.

[45] B. Matérn, *Spatial Variation*, 2nd ed., Lecture Notes in Statist. 36, Springer, Berlin, 1986.

[46] G. Matheron, *The Theory of Regionalized Variables and its Applications*, Ecole de Mines, Fontainebleau, 1971.

[47] V. Minden, A. Damle, K. L. Ho and L. Ying, Fast spatial Gaussian process maximum likelihood estimation via skeletonization factorizations, *Multiscale Model. Simul.* **15** (2017), no. 4, 1584–1611.

[48] W. G. Müller, *Collecting Spatial Data. Optimum Design of Experiments for Random Fields*, 3rd ed., Contrib. Statist., Springer, Berlin, 2007.

[49] G. R. North, J. Wang and M. G. Genton, Correlation models for temperature fields, *J. Climate* **24** (2011), 5850–5862.

[50] W. Nowak, Measures of parameter uncertainty in geostatistical estimation and geostatistical optimal design, *Math. Geosci* **42** (2010), no. 2, 199–221.

[51] W. Nowak and A. Litvinenko, Kriging and spatial design accelerated by orders of magnitude: Combining low-rank covariance approximations with FFT-techniques, *Math. Geosci.* **45** (2013), no. 4, 411–435.

[52] D. Nychka, S. Bandyopadhyay, D. Hammerling, F. Lindgren and S. Sain, A multiresolution Gaussian process model for the analysis of large spatial datasets, *J. Comput. Graph. Statist.* **24** (2015), no. 2, 579–599.

[53] I. V. Oseledets, Tensor-train decomposition, *SIAM J. Sci. Comput.* **33** (2011), no. 5, 2295–2317.

[54] J. Quiñonero Candela and C. E. Rasmussen, A unifying view of sparse approximate Gaussian process regression, *J. Mach. Learn. Res.* **6** (2005), 1939–1959.

[55] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, Adapt. Comput. Mach. Learn., MIT, Cambridge, 2006.

[56] A. K. Saibaba, S. Ambikasaran, J. Yue Li, P. K. Kitanidis and E. F. Darve, Application of hierarchical matrices to linear inverse problems in geostatistics, *Oil Gas Sci. Technol. Rev. IFP Energ. Nouv.* **67** (2012), no. 5, 857–875.

[57] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, *Ann. Physics* **326** (2011), no. 1, 96–192.

[58] R. Shah and P. Reed, Comparative analysis of multiobjective evolutionary algorithms for random and correlated instances of multiobjective *d*-dimensional knapsack problems, *European J. Oper. Res.* **211** (2011), no. 3, 466–479.

[59] A. K. Smilde, R. Bro and P. Geladi, *Multi-Way Analysis with Applications in the Chemical Sciences*, Wiley, New York, 2004.

[60] G. Spöck and J. Pilz, Spatial sampling design and covariance-robust minimax prediction based on convex design ideas, *Stoch. Environmental Res. Risk Assess.* **24** (2010), 463–482.

[61] M. L. Stein, J. Chen and M. Anitescu, Difference filter preconditioning for large covariance matrices, *SIAM J. Matrix Anal. Appl.* **33** (2012), no. 1, 52–72.

[62] M. L. Stein, Z. Chi and L. J. Welty, Approximating likelihoods for large spatial data sets, *J. R. Stat. Soc. Ser. B Stat. Methodol.* **66** (2004), no. 2, 275–296.

[63] F. Stenger, *Numerical Methods Based on Sinc and Analytic Functions*, Springer Ser. Comput. Math. 20, Springer, New York, 1993.

[64] Y. Sun and M. L. Stein, Statistically and computationally efficient estimating equations for large spatial datasets, *J. Comput. Graph. Statist.* **25** (2016), no. 1, 187–208.

[65] L. R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika* **31** (1966), 279–311.

[66] S. M. Wesson and G. G. S. Pegram, Radar rainfall image repair techniques, *Hydrol. Earth Syst. Sci.* **8** (2004), no. 2, 8220–8234.