POLITECNICO DI TORINO
Repository ISTITUZIONALE

RBF-based tensor decomposition with applications to oenology

Original
RBF-based tensor decomposition with applications to oenology / Perracchione, E.. - In: DOLOMITES RESEARCH NOTES ON APPROXIMATION. - ISSN 2035-6803. - 13:(2020), pp. 36-46. [10.14658/PUPJ-DRNA-2020-1-5]

Availability:
This version is available at: 11583/2987673 since: 2024-04-09T08:40:55Z

Publisher:
Padova University Press

Published
DOI:10.14658/PUPJ-DRNA-2020-1-5

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

29 June 2026

Dolomites Research Notes on Approximation

Volume 13 · 2020 · Pages 36–46

RBF-based tensor decomposition with applications to oenology

Emma Perracchione[a]

Communicated by S. De Marchi

Abstract

As usually claimed, meshless methods work in any dimension and are easy to implement. However in practice, to preserve the convergence order when the dimension grows, they need a huge number of sampling points and both computational costs and memory turn out to be prohibitive. Moreover, when a large number of points is involved, the usual instability of the Radial Basis Function (RBF) approximants becomes evident. To partially overcome this drawback, we propose to apply *tensor decomposition methods*. This, together with *rational* RBFs, allows us to obtain efficient interpolation schemes for high dimensions. The effectiveness of our approach is also verified by an application to oenology.

1 Introduction

The main topic of the proposed study focuses on the approximation of multivariate functions. Such problem naturally entails a large number of intimately related computational and theoretical issues, such as overcoming the curse of dimensionality, that need further investigations from the scientific community. Indeed, the multivariate approximation finds many applications; to mention a few, reconstruction of medical images and approximation of multivariate PDEs [14, 25]. Moreover, it is common to all numerical approximation schemes, as polynomial expansions, spline interpolation and meshfree or meshless methods. For the latter, except for particular instances, see e.g. [10, 33], the numerical tests are usually carried out in low dimensions.

When the dimension grows, a possible solution for overcoming the huge computational complexity comes from the use of local schemes, as the Partition of Unity (PU) method; refer e.g. to [11, 37]. However, this might not be completely satisfying. Indeed, for constructing the local subdomains, large grids in high dimensions need to be computed, leading again to prohibitive computational costs.

Here we propose a tool that enables us to effectively interpolate in high dimensions and that comes from the field of Tensor Decomposition (TD) methods. Indeed, for modeling multivariate phenomena, Reduced Order Methods (ROMs), known as Karhunen-Loève expansion (KLE) have gained popularity [26]. We focus on the Proper Orthogonal Decomposition (POD) [1, 6], which also finds many applications in computational fluid dynamics [4, 28, 35]. As the Higher Order SVD (HOSVD) [13], it provides a multilinear generalization of the best low rank approximation problem for matrices obtained by truncating the SVD. In fact, the HOSVD enables us to perform a low dimensional approximation of tensors as the SVD allows to approximate bivariate data. Moreover, the POD takes advantage of defining a clear theoretical framework and it is based on the famed Mercer's eigenfunctions, which have also been used to compute stable bases for kernel-based approximants via low rank representations of the interpolation matrix [18].

To summarize, TD tools allow us to consider univariate RBF interpolation in each direction, producing effective approximations. As RBF-bases, we also consider the so-called eigen-rational kernels [8] and we study properties of the spectrum of the rational kernel matrices.

The paper is organized as follows. In Section 2, we present the well-known RBF theory for multivariate interpolation and we also introduce the rational approximants that can be seen as Hadamard product of standard kernels (see e.g. [34]). Then, in Section 3, we describe our algorithm, namely in what follows TD-RBF, based on univariate kernel approximation via POD. Section 4 is devoted to numerical experiments. Applications to wine preferences, a typical example of multivariate modeling, are presented in Section 5. We conclude with some final remarks concerning possible extensions of the present approach.

2 Rational RBF-based interpolation

In the next subsection we review the main theoretical aspects of kernel-based methods and in doing so we mainly refer to the books [7, 20, 21, 36]. Then we present the rational bases, first introduced in [24] and then further investigated in [8, 17, 30, 31].

2.1 Remarks on kernel-based interpolation

Given $\mathcal{X}_N = \{\boldsymbol{x}_i,\ i = 1,\dots,N\} \subset \Omega$ a set of distinct data points (or data sites or nodes), arbitrarily distributed on a domain $\Omega \subseteq \mathbb{R}^M$, and the associated set $\mathcal{F}_N = \{f_i = f(\boldsymbol{x}_i),\ i = 1,\dots,N\}$ of data values (or measurements or function values), the scattered

[a]Department of Mathematics DIMA, University of Genova; perracchione@dima.unige.it

data interpolation problem consists in finding a function $P_f : \Omega \longrightarrow \mathbb{R}$ such that

$$P_f(\boldsymbol{x}_i) = f_i, \quad i = 1, \dots, N.$$

Here we focus on RBF interpolants $P_f : \Omega \longrightarrow \mathbb{R}$ which assume the form [20]

$$P_f(\boldsymbol{x}) = \sum_{k=1}^{N} \alpha_k K(\boldsymbol{x}, \boldsymbol{x}_k), \quad \boldsymbol{x} \in \Omega,$$

where K is a strictly positive definite kernel. To such kind of kernels, we associate a univariate function $\phi : [0, \infty) \to \mathbb{R}$ (possibly depending on a shape parameter $\varepsilon > 0$), such that

$$K(\boldsymbol{x}, \boldsymbol{y}) = \phi_\varepsilon(||\boldsymbol{x} - \boldsymbol{y}||_2) := \phi(r), \quad \boldsymbol{x}, \boldsymbol{y} \in \Omega.$$

By imposing the interpolation conditions, the scattered data interpolation problem reduces to solving a linear system of the form

$$\mathsf{K}\boldsymbol{\alpha} = \boldsymbol{f} \tag{1}$$

where

$$\mathsf{K}_{ik} = K(\boldsymbol{x}_i, \boldsymbol{x}_k), \quad i, k = 1, \dots, N,$$

$\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_N)^\intercal$ and $\boldsymbol{f} = (f_1, \dots, f_N)^\intercal$. Under our assumptions, such a system admits a unique solution [20].

Concerning error bounds, we start by associating to K a real pre-Hilbert space $H_K(\Omega)$ with reproducing kernel K (refer e.g. to [36])

$$H_K(\Omega) = \text{span}\{K(\cdot, \boldsymbol{x}),\ \boldsymbol{x} \in \Omega\},$$

equipped with the bilinear form $(\cdot, \cdot)_{H_K(\Omega)}$. We then define the *native space* $\mathcal{N}_K(\Omega)$ of K as the completion of $H_K(\Omega)$ with respect to the norm $||\cdot||_{H_K(\Omega)}$, that is $||f||_{H_K(\Omega)} = ||f||_{\mathcal{N}_K(\Omega)}$ for all $f \in H_K(\Omega)$.

We point out that a pointwise error bound for kernel-based interpolants is of the form [21]

$$|f(\boldsymbol{x}) - P_f(\boldsymbol{x})| \leq C h^\beta ||f||_{\mathcal{N}_K(\Omega)}, \quad \boldsymbol{x} \in \Omega,$$

for some appropriate exponent β depending on the smoothness of the kernel and h is the so-called fill distance given by:

$$h = \sup_{\boldsymbol{x} \in \Omega} \left(\min_{\boldsymbol{x}_k \in \mathcal{X}_N} \|\boldsymbol{x} - \boldsymbol{x}_k\|_2 \right).$$

The fill distance and hence the error depends on the number of data N. The latter, to overcome the curse of dimensionality, must grow according to M. To better understand this fact, supposing quasi-uniformly distributed data sites, i.e., $h = \mathcal{O}(N^{-1/M})$, we have that

$$|f(\boldsymbol{x}) - P_f(\boldsymbol{x})| \leq C_M N^{-\beta/M} ||f||_{\mathcal{N}_K(\Omega)}, \quad \boldsymbol{x} \in \Omega,$$

which shows that the rate of convergence deteriorates as M increases and where C_M depends on the dimension.

Thus, our main objective is the one of preserving the accuracy also in higher dimensions. To such aim we introduce *tensor decomposition methods*. Before doing this, we present a recent tool, namely the eigen-rational interpolant, which offers a different and possibly more accurate computation of the RBF-based interpolant.

2.2 Eigen-rational interpolants as Hadamard products

Focusing on strictly positive definite kernels K, the eigen-rational interpolant assumes the form

$$\hat{P}_f(\boldsymbol{x}) = \frac{\sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i)}{\sum_{k=1}^{N} \beta_k K(\boldsymbol{x}, \boldsymbol{x}_k)} = \frac{P_g(\boldsymbol{x})}{P_h(\boldsymbol{x})}, \tag{2}$$

defined for some function values g_i, h_i, $i = 1, \dots, N$ [8]. The coefficients $\boldsymbol{\beta} = (\beta_1, \dots, \beta_N)^\intercal$ are selected so that $\boldsymbol{\beta}$ is the eigenvector associated to the largest eigenvalue of K. As shown in [8], this makes the problem well-defined. Indeed, once we compute the vector $\boldsymbol{\beta}$, we can calculate the function values $P_h(\boldsymbol{x}_i) = h_i$, $i = 1, \dots, N$. Then, we construct the eigen-rational interpolant $\hat{P}_f$ by solving (1), where the vector of function values is replaced by $\boldsymbol{g} = \boldsymbol{f}\boldsymbol{h}$.

We remark that the eigen-rational interpolant shows similar convergence rates to the ones of the standard interpolant. However, numerically, we can register a sensible improvement in terms of stability. This results in an effective method for the approximation problem.

We here investigate a few properties concerning the spectrum of the eigen-rational interpolant compared to the standard kernel matrix (see [19]). To this aim we report a result by Schur [34] useful for that purposes (refer also to [23, Lemma A.5] and [22, Lemma 2.1]).

Theorem 2.1. *When* E *and* $\mathsf{M} \in \mathbb{R}^{N \times N}$ *are positive definite matrices, denoting by* $\lambda_{\min}$ *and* $\lambda_{\max}$ *the smallest and largest eigenvalue of a matrix, we have that*

$$\lambda_{\min}(\mathsf{E}) \min_{i=1,\dots,N}(\mathsf{M}_{ii}) \leq \lambda_i(\mathsf{E} \circ \mathsf{M}) \leq \lambda_{\max}(\mathsf{E}) \max_{i=1,\dots,N}(\mathsf{M}_{ii}),$$

where $\circ$ *is the Hadamard product, i.e.*

$$(\mathsf{E} \circ \mathsf{M})_{ij} = \mathsf{E}_{ij}\mathsf{M}_{ij},$$

$i, j = 1, \dots, N$.

In [8] the authors proved that the interpolant defined in (2) can be equivalently written via the rational kernel:

$$K_R(\boldsymbol{x},\boldsymbol{y}) = K(\boldsymbol{x},\boldsymbol{y})\frac{1}{P_h(\boldsymbol{x})}\frac{1}{P_h(\boldsymbol{y})}.$$

Note that K_R is strictly positive definite provided so is K. In view of this we have the following result.

Property 2.1. Suppose K is a strictly positive definite kernel, then the rational kernel matrix denoted by K^R assumes the form

$$\mathsf{K}^R = \mathsf{K} \circ \hat{\mathsf{K}},$$

where

$$\hat{\mathsf{K}}_{ij} = \frac{1}{P_h(\boldsymbol{x}_i)}\frac{1}{P_h(\boldsymbol{x}_j)}, \quad i,j = 1,\dots,N.$$

Moreover,

$$\lambda_{\min}(\mathsf{K}) \min_{i=1,\dots,N}(\hat{\mathsf{K}}_{ii}) \le \lambda_i(\mathsf{K}\circ\hat{\mathsf{K}}) \le \lambda_{\max}(\mathsf{K}) \max_{i=1,\dots,N}(\hat{\mathsf{K}}_{ii}),$$

for $i = 1,\dots,N$.

Proof. Since $h_j = \sum_{i=1}^N \beta_i K(\boldsymbol{x}_j,\boldsymbol{x}_i)$, we have that

$$\hat{P}_f = \sum_{j=1}^N \alpha_j \frac{K(\boldsymbol{x},\boldsymbol{x}_j)}{\sum_{i=1}^N \beta_i K(\boldsymbol{x},\boldsymbol{x}_i)} = \sum_{j=1}^N \alpha_j \frac{h_j K(\boldsymbol{x},\boldsymbol{x}_j)}{\sum_{i=1}^N \beta_i K(\boldsymbol{x},\boldsymbol{x}_i)\sum_{i=1}^N \beta_i K(\boldsymbol{x}_j,\boldsymbol{x}_i)}.$$

Thus,

$$\hat{P}_f = \sum_{j=1}^N \tilde{\alpha}_j \frac{K(\boldsymbol{x},\boldsymbol{x}_j)}{\sum_{i=1}^N \beta_i K(\boldsymbol{x},\boldsymbol{x}_i)\sum_{i=1}^N \beta_i K(\boldsymbol{x}_j,\boldsymbol{x}_i)} =: \sum_{j=1}^N \tilde{\alpha}_j K_R(\boldsymbol{x},\boldsymbol{x}_j).$$

Trivially,

$$\mathsf{K}^R = \mathsf{K} \circ \hat{\mathsf{K}},$$

and the thesis follows from Theorem 2.1 and by [27]. □

In [8], the authors show that the *rational* RBF interpolant (called in what follows RRBF) is a rescaled version of the standard one. Thus, the above property confirms that the spectrum of the rational kernel is also a rescaled form of the standard one.

3 RBF-based POD approach

In order to present the POD expansion for M-variate functions, with $M \ge 3$, (see e.g. [2, 3]), we need to point out a few remarks on the POD for bivariate functions in the next subsection; refer e.g. to [1].

3.1 Remarks on the POD

Given a function $f \in L^2(X_1 \times X_2)$, where here $X_1, X_2 \subseteq \mathbb{R}$, we are interested in the Hilbert-Schmidt integral operator with kernel f given by

$$(Y\varphi)(x_2) = \int_{X_1} f(x_1,x_2)\varphi(x_1)dx_1,$$

where $\varphi \in L^2(X_1)$ and in its adjoint $Y^* : L^2(X_2) \to L^2(X_1)$, defined as

$$(Y^* v)(x_1) = \int_{X_2} f(x_1,x_2)v(x_2)dx_2,$$

with $v \in L^2(X_2)$. The function f is thus the kernel of the self adjoint operator $P = Y^*Y$ and, by virtue of the Mercer's theorem [29], it admits the following convergent expansion in $L^2(X_1 \times X_2)$, see e.g. [1, Corollary 2.2]:

$$f(x_1,x_2) = \sum_{i\in\mathbb{N}} \sigma_i \varphi_i(x_1) v_i(x_2),$$

with $v \in L^2(X_2)$ and where the eigenfunctions $(\varphi_i)_{i\in\mathbb{N}}$ and associated eigenvalues $(\lambda_i)_{i\in\mathbb{N}}$ of P form a complete orthonormal basis of $L^2(X_1)$. We assume that the eigenvalues are ordered in decreasing values. As a consequence, the sequence $(v_i)_{i\in\mathbb{N}}$ is an orthogonal basis of $L^2(X_2)$ and it is defined as

$$v_i = \frac{1}{\sigma_i} Y\varphi_i,$$

where $\sigma_i = \sqrt{\lambda_i}$ are the so-called singular values and $\varphi_i = Y^* v_i$.

Note that in the discrete case, i.e. when we have samples of function values, this is indeed equivalent to the SVD, as the discrete Recursive POD (RPOD) that we are going to describe can be seen as a higher dimensional SVD.

3.2 Multivariate POD expansion and kernel interpolation

We now suppose that f is in the Lebesgue space $L^2(X_1 \times X_2 \times \cdots \times X_M)$, where $X_1, \ldots, X_M \subset \mathbb{R}$ are bounded domains. Then, in view of the above considerations, we have

$$f(x_1, x_2, \ldots, x_M) = \sum_{i_1 \in \mathbb{N}} \sigma_{i_1} v_{i_1}(x_2, \ldots, x_M) \varphi_{i_1}(x_1),$$

where the sum is convergent in $L^2(X_2 \times \cdots \times X_M, L^2(X_1))$ and where $(\varphi_{i_1})_{i_1 \in \mathbb{N}}$ and $(v_{i_1})_{i_1 \in \mathbb{N}}$, are two orthonormal sets respectively complete in $L^2(X_1)$ and $L^2(X_2 \times \cdots \times X_M)$.

Continuing in applying recursively the POD, we construct the expansion of

$$v_{i_1}, v_{i_2}^{(i_1)}, \ldots, v_{i_{M-2}}^{(i_1, i_2, \ldots, i_{M-3})},$$

so that $f \in L^2(X_1 \times X_2 \times \cdots \times X_M)$ can be written as [3, Lemma 2.1]

$$\begin{aligned} f &= \sum_{i_1 \in \mathbb{N}} \sum_{i_2 \in \mathbb{N}} \cdots \sum_{i_{M-1} \in \mathbb{N}} \sigma_{i_1} \sigma_{i_2}^{(i_1)} \cdots \sigma_{i_{M-1}}^{(i_1, i_2, \cdots, i_{M-2})} \varphi_{i_1} \otimes \varphi_{i_2}^{(i_1)} \otimes \cdots \otimes \varphi_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})} \otimes v_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})}, \\ &= \sum_{i_1 \in \mathbb{N}} \sigma_{i_1} \Bigg(\sum_{i_2 \in \mathbb{N}} \sigma_{i_2}^{(i_1)} \cdots \Bigg(\sum_{i_{M-1} \in \mathbb{N}} \sigma_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})} \varphi_{i_1} \otimes \varphi_{i_2}^{(i_1)} \Bigg) \otimes \cdots \otimes \varphi_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})} \Bigg) \otimes v_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})}, \end{aligned} \tag{3}$$

where sums are convergent in $L^2(X_1 \times X_2 \times \cdots \times X_M)$.

Computationally speaking this can be seen as a SVD in higher dimensions and thus it might be useful studying criteria for truncating the expansion as

$$f_T = \sum_{i_1=0}^{I_1} \sigma_{i_1} \Bigg(\sum_{i_2=0}^{I_2^{(i_1)}} \sigma_{i_2}^{(i_1)} \cdots \Bigg(\sum_{i_{M-1}=0}^{I_{M-1}^{(i_1, i_2, \ldots, i_{M-2})}} \sigma_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})} \varphi_{i_1} \otimes \varphi_{i_2}^{(i_1)} \Bigg) \otimes \cdots \otimes \varphi_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})} \Bigg) \otimes v_{i_{M-1}}^{(i_1, i_2, \ldots, i_{M-2})}.$$

We use a multi-index notation with T that stands for

$$T = \left(I_1, I_2^{(i_1)}, \ldots, I_{M-1}^{(i_1, \ldots, i_{M-2})} \right),$$

where $T_i \geq 1$, $i = 1, \ldots, M-1$.

Concerning the truncation, we have the following result, see [3, Lemma 5.5].

Lemma 3.1. *Assume that the coefficients of the RPOD expansion (3) satisfy*

$$\sum_{i_1 > I_1} \left| \sigma_{i_1} \right|^2 \leq \Sigma_1, \quad \sum_{i_k > I_k^{(i_1, \ldots, i_{k-1})}} \left| \sigma_{i_k}^{(i_1, \ldots, i_{k-1})} \right|^2 \leq \Sigma_k, \quad k = 2, \ldots, M-1. \tag{4}$$

Then, the following error estimate in L^2 norm holds

$$\left|\left| f - f_T \right|\right|^2_{L^2(X_1 \times X_2 \times \cdots \times X_M)} \leq \sigma_{I_1+1} + C_T \left|\left| f \right|\right|^2_{L^2(X_1 \times X_2 \times \cdots \times X_M)}, \tag{5}$$

where $C_T = \Sigma_2 + \Sigma_{M-2} + \Sigma_{M-1}$, with

$$\Sigma_k = \max_{i_1, \ldots, i_{k-1}} \sigma_{I_k+1}^{(i_1, \ldots, i_{k-1})} \quad \text{and} \quad I_k = \min_{i_1, \ldots, i_{k-1}} I_k^{(i_1, \ldots, i_{k-1})},$$

$k = 2, \ldots, M-1$.

Proof.

$$\begin{aligned} ||f - f_T||^2_{L^2(X_1 \times X_2 \times \cdots \times X_M)} =& \Bigg(\sum_{i_1=0}^{I_1} \left| \sigma_{i_1} \right|^2 \sum_{i_2=0}^{I_2^{(i_1)}} \left| \sigma_{i_2}^{(i_1)} \right|^2 \cdots \sum_{i_{M-2}=0}^{I_{M-2}^{(i_1, \ldots, i_{M-3})}} \left| \sigma_{i_{M-2}}^{(i_1, \ldots, i_{M-3})} \right|^2 \sum_{i_{M-1} > I_{M-1}^{(i_1, \ldots, i_{M-2})}} \left| \sigma_{i_{M-1}}^{(i_1, \ldots, i_{M-2})} \right|^2 \Bigg) \\ &+ \Bigg(\sum_{i_1=0}^{I_1} \left| \sigma_{i_1} \right|^2 \sum_{i_2=0}^{I_2^{(i_1)}} \left| \sigma_{i_2}^{(i_1)} \right|^2 \cdots \sum_{i_{M-2} > I_{M-2}^{(i_1, \ldots, i_{M-3})}} \left| \sigma_{i_{M-2}}^{(i_1, \ldots, i_{M-3})} \right|^2 \Bigg) + \cdots \\ &+ \Bigg(\sum_{i_1=0}^{I_1} \left| \sigma_{i_1} \right|^2 \sum_{i_2 > I_2^{(i_1)}} \left| \sigma_{i_2}^{(i_1)} \right|^2 \Bigg) + \sum_{i_1 > I_1} \left| \sigma_{i_1} \right|^2. \end{aligned}$$

Using (4), we deduce that

$$\begin{aligned}||f-f_T||^2_{L^2(X_1\times X_2\times\cdots\times X_M)} \leq & \Sigma_{M-1}\Bigg(\sum_{i_1=0}^{I_1}\left|\sigma_{i_1}\right|^2\sum_{i_2=0}^{I_2^{(i_1)}}\left|\sigma_{i_2}^{(i_1)}\right|^2\cdots\sum_{i_{M-2}=0}^{I_{M-2}^{(i_1,\ldots,i_{M-3})}}\left|\sigma_{i_{M-2}}^{(i_1,\ldots,i_{M-3})}\right|^2\Bigg)\\ & +\Sigma_{M-2}\Bigg(\sum_{i_1=0}^{I_1}\left|\sigma_{i_1}\right|^2\sum_{i_2=0}^{I_2^{(i_1)}}\left|\sigma_{i_2}^{(i_1)}\right|^2\cdots\sum_{i_{M-3}=0}^{I_{M-3}^{(i_1,\ldots,i_{M-4})}}\left|\sigma_{i_{M-3}}^{(i_1,\ldots,i_{M-4})}\right|^2\Bigg)+\cdots\\ & +\Sigma_2\sum_{i_1=0}^{I_1}\left|\sigma_{i_1}\right|^2+\sum_{i_1>I_1}\left|\sigma_{i_1}\right|^2\leq C_T\sum_{i_1=0}^{I_1}\left|\sigma_{i_1}\right|^2+\sigma_{I_1+1}.\end{aligned}$$

Then, since

$$\sum_{i_1=0}^{I_1}\left|\sigma_{i_1}\right|^2\leq\left|\left|f\right|\right|^2_{L^2(X_1\times X_2\times\cdots\times X_M)},$$

(5) follows. □

In what follows, even if we will decompose the discrete tensor for interpolation without truncation, as explained in [3], such estimate allows to derive a practical strategy for the computation of the truncated RPOD expansion. Moreover, such re-ordered POD expansion turns out to be handy from a theoretical point of view and easy to implement. However, we suggest, for the specific problem of reconstructing function values, the use of the multi-index notation. In fact, even if the two representations are equivalent, i.e. the number of modes does not change, the use of the multi-index notation allows to factor out several eigenfunctions for the interpolation process and this surely leads to a cipher procedure (see the `TD-RBF Algorithm`). In such algorithm we summarize the steps of the procedure used in our experiments. In the pseudocode, we denote with "Interp" a general interpolation procedure that corresponds to either rational or standard RBF bases.

`INPUTS:` The tensor constructed on the set $\mathcal{X}_N$, i.e. the set $\mathcal{F}_N$;
the evaluation point $\boldsymbol{x}^*\in\Omega$.

`OUTPUTS:` $f(\boldsymbol{x}^*)$

`Step 1:` Set $f(\boldsymbol{x}^*)=0$
For $i_1=0:I_1$
$\varphi_{i_1}(x_1^*)=\text{Interp}(\varphi_{i_1},x_1^*)$

`Step 2:` **For** $i_2=0:I_2^{(i_1)}$
$\varphi_{i_2}^{(i_1)}(x_2^*)=\text{Interp}(\varphi_{i_2}^{(1_1)},x_2^*)$
$\vdots$

`Step M:` **For** $i_{M-1}^{(i_1,i_2,\ldots,i_{M-2})}=0:I_{M-1}^{(i_1,i_2,\ldots,i_{M-2})}$
$v_{i_{M-1}}^{(i_1,i_2,\ldots,i_{M-2})}(x_M^*)=\text{Interp}(v_{i_{M-1}}^{(i_1,i_2,\ldots,i_{M-2})},x_M^*)$
$\rho=\sigma_{i_1}\sigma_{i_2}^{(i_1)}\ldots\sigma_{i_{M-1}}^{(i_1,i_2,\ldots,i_{M-2})}$

`Step M+1:`
$f(\boldsymbol{x}^*)=f(\boldsymbol{x}^*)+\rho\varphi_{i_1}(x_1^*)\varphi_{i_2}^{(i_1)}(x_2^*)\ldots\varphi_{i_{M-1}}^{(i_1,i_2,\ldots,i_{M-2})}(x_{M-1}^*)v_{i_{M-1}}^{(i_1,i_2,\ldots,i_{M-2})}(x_M^*)$

The `TD-RBF Algorithm`. The TD-RBF pseudocode used to estimate function values.

4 Numerical experiments

We compare standard multivariate RBF interpolation, namely in what follows ST-RBF, with the algorithm TD-RBF, eventually with the use of RRBFs.

The numerical tests are carried out in dimension $M=2,3,4,5$. They are devoted to point out that:

1. the instability of the classical RBF interpolation grows according to the dimension M and in those cases the TD-RBF becomes essential;
2. for high dimensions the computational effort is prohibitive without the TD-RBF scheme;
3. the TD-RRBF allows to substantially increase the accuracy, especially for the Gaussian kernel.

In this section, we construct tensors by sampling several functions on $(2k)^M$ grid data, with $M = 2,\dots,5$, and $k = 1,\dots,5$, on $\Omega = [0,1]^M$. We then evaluate the reconstruction on 100^M random data $\bar{\boldsymbol{x}}_i, i = 1,\dots,100^M$, by computing the Relative Maximum Absolute Error and the Relative Root Mean Square Error, respectively given by

$$\text{RMAE} = \max_{x\in\Omega} \frac{|f(\boldsymbol{x}) - A(\boldsymbol{x})|}{|f(\boldsymbol{x})|},$$

and

$$\text{RRMSE} = \frac{||f(\boldsymbol{x}) - A(\boldsymbol{x})||_2}{||f(\boldsymbol{x})||_2},$$

where A in an approximant constructed via ST-RBF, TD-RBF or TD-RRBF. For all the experiments we take the Gaussian kernel.

At first we focus on the items 1. and 2. of the above list. To such aim, we take the following test function [20]:

$$f_1(\boldsymbol{x}) = 16\prod_{i=1}^{M} x_i(1-x_i).$$

The results of using ST-RBF and TD-RBF are plotted in Figure 1. The shape parameter for both cases is fixed as $\varepsilon = 0.8$. We can note that, as expected, in the 2-dimensional framework the TD-RBF is meaningful only if a large number of points is involved. Conversely, in higher dimensions the instability becomes evident and decomposing the tensor enables us to mitigate such effect. Concerning the computational complexity, for $M = 5$ the memory requirement for the ST-RBF becomes prohibitive for 6^M points. Experiments are carried out with MATLAB on an Intel(R) Core(TM) i7 CPU 4712MQ 2.13 GHz processor.

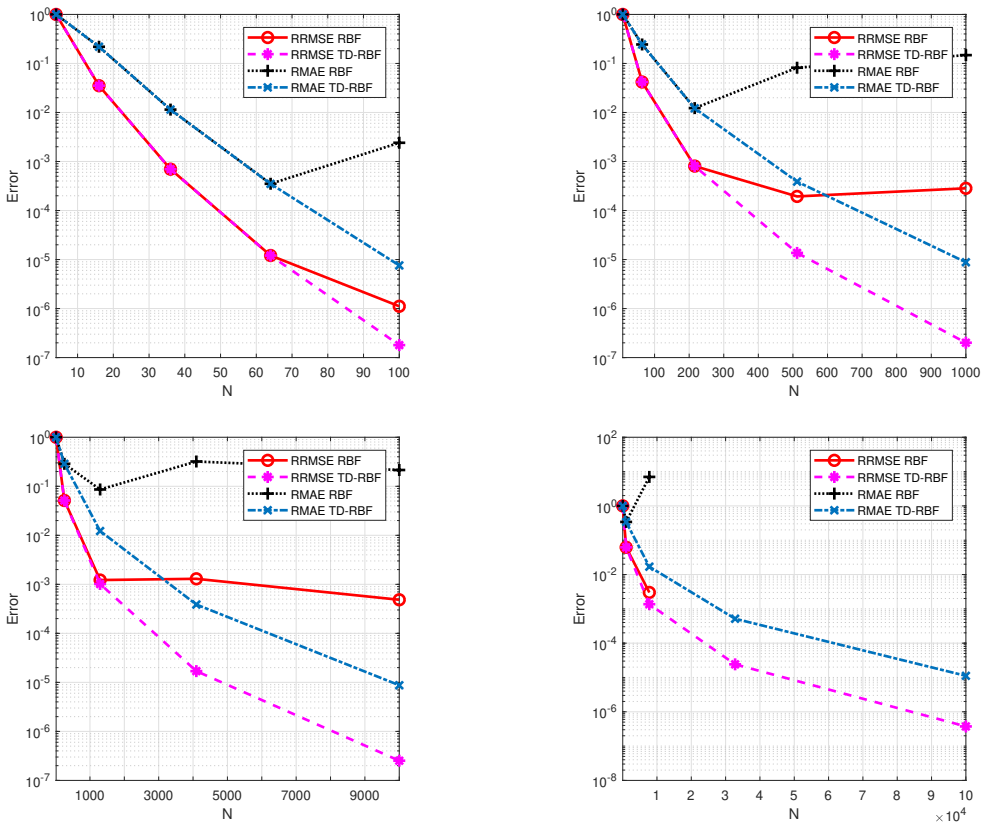

Figure 1: Number of points VS RMAE and RRMSE for ST-RBF and TD-RBF with the function f_1. From top to bottom, left to right, $M = 2$, 3, 4 and 5.

We now focus on the possible benefits coming from the use of rational bases. In doing so, we fix $\varepsilon = 1$ and we consider different test functions reported below for simplicity:

$$f_{2,2}(\boldsymbol{x}) = \frac{\text{sinc}(x_1)x_2^2 + \text{e}^{x_1}}{1 + x_1 + x_2^2}, \quad f_{2,3}(\boldsymbol{x}) = \frac{\text{sinc}(x_1)\text{sinc}(x_3)x_2^2 + \text{e}^{x_1}}{1 + x_1 + x_2^2 + x_3^3},$$

$$f_{2,4}(\boldsymbol{x}) = \frac{\text{sinc}(x_1)\text{sinc}(x_3)x_2^2x_4^2 + \text{e}^{x_1}}{1 + x_1 + x_2^2 + x_3^3 - x_4^4}, \quad f_{2,5}(\boldsymbol{x}) = \frac{\text{sinc}(x_1)\text{sinc}(x_3)\text{sinc}(x_4)x_2^2x_4^2 + \text{e}^{x_1}}{1 + x_1 + x_2^2 + x_3^3 + x_4^4 + x_5^5},$$

Table 2: Condition numbers of kernel matrices.

	$M>1$		TD	
N	cond(K)	cond(K^R)	cond(K)	cond(K^R)
2^M	1.01E+01	1.01E+01	2.16E+00	2.16E+00
4^M	1.32E+09	1.87E+09	1.09E+03	1.35E+03
6^M	8.04E+17	7.16E+17	3.38E+06	4.14E+06
8^M	1.90E+19	3.50E+19	2.52E+10	3.07E+10
10^M	2.09E+20	3.64E+20	3.41E+14	4.12E+14

$$f_{3,2}(\boldsymbol{x}) = \cos(x_1) + -\log(x_1x_2+7), \quad f_{3,3}(\boldsymbol{x}) = \cos(x_1) + \cos(x_3) - \log(x_1x_2x_3+7),$$
$$f_{3,4}(\boldsymbol{x}) = \cos(x_1) + \cos(x_3) - \log(x_1x_2x_3x_4+7), \quad f_{3,5}(\boldsymbol{x}) = \cos(x_1) + \cos(x_3) - \log(x_1x_2x_3x_4x_5+7).$$

Before going into details and comparing ST-RBF with TD-RRBF, we report the condition numbers of the kernel interpolation matrices. Precisely, in Table 2, we show the condition numbers of the matrices K and K^R. As expected, being the spectrum of the RRBFs a rescaled version of the one obtained via RBFs, the condition numbers are comparable. Of course, being the matrices dependent on the distance among points, the results are analogous for each $M>1$. While, in the last two columns we report the condition numbers for the univariate interpolation matrices involved in the computation after decomposing the tensors.

The results of using $f_{2,M}$, $M=2,\dots,5$ are shown in Figure 2. We can note that in this case the use of rational bases is particularly meaningful in higher dimensions. Such improvement is even more evident with the second set of test functions; refer to Figure 3. We conclude that, if in addition to the tensor decomposition one also uses rational bases, the proposed tool turns out to be robust and stable. As a confirm, we consider in the next section an application to wine preferences.

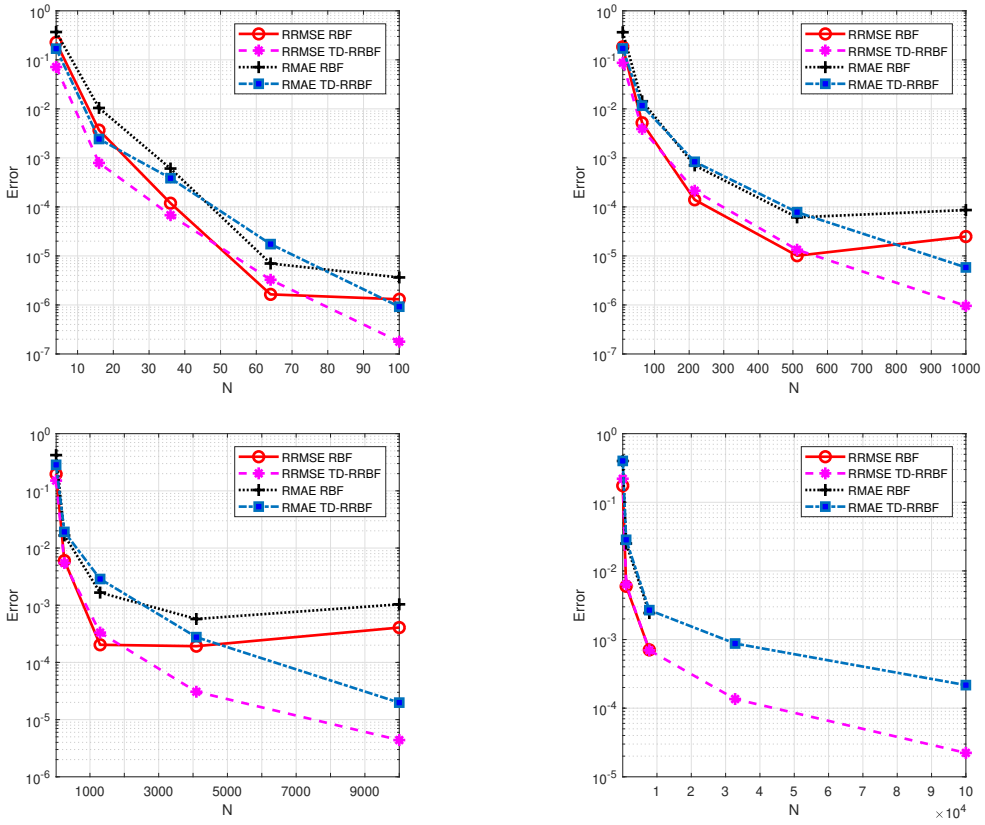

Figure 2: Number of points VS RMAE and RRMSE for ST-RBF and TD-RRBF with the function $f_{2,M}$. From top to bottom, left to right, $M=2$, 3, 4 and 5.

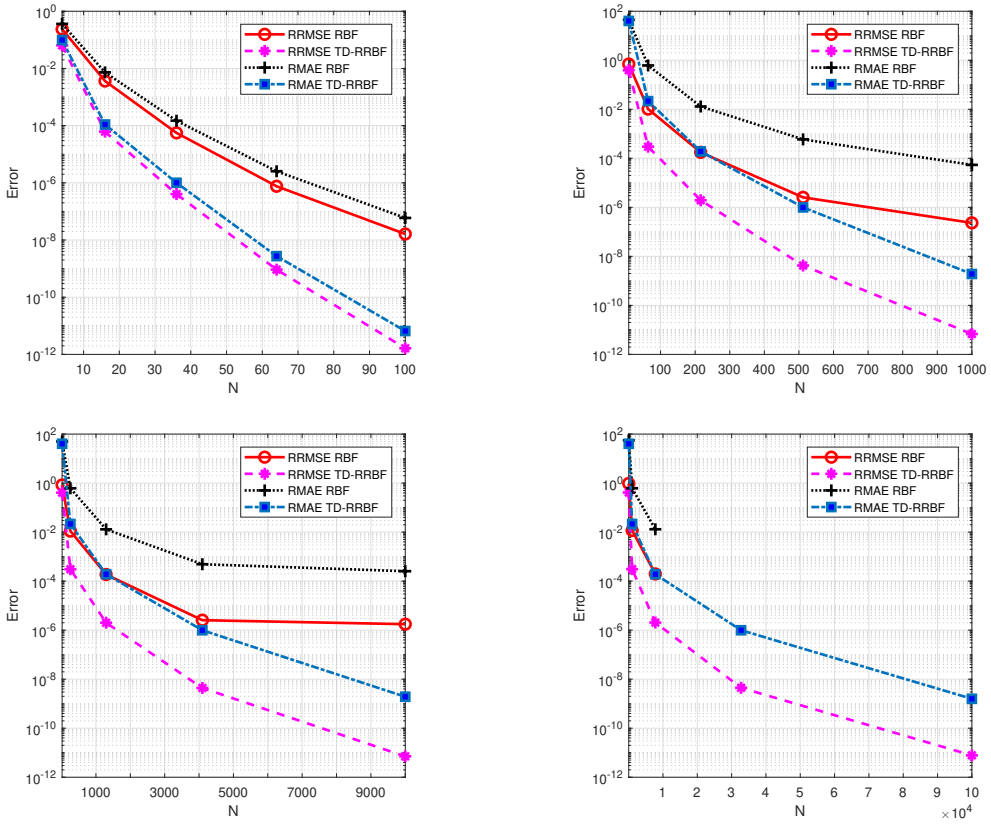

Figure 3: Number of points VS RMAE and RRMSE for ST-RBF and TD-RRBF with the function $f_{3,M}$. From top to bottom, left to right, $M = 2$, 3, 4 and 5.

Table 3: Numerical simulations for the red wine data.

Method	ST-RBF	TD-RRBF
RMAE	8.71E+00	6.66E−01
RRMSE	1.24E+00	1.28E−01

5 Applications to wine preferences

As fancy application, we consider the problem of scoring wines based on several input parameters. This is indeed, a typical example of multivariate data analysis. The data set we consider is available at: https://archive.ics.uci.edu/ml/datasets/wine and has been created in [12]. This is frequently used for testing both classification (support vector machine) and regression (support vector regression) algorithms. However, due to the high number of input parameters the problem becomes challenging and we here apply our tensor decomposition tool. The 3^{11} toy tensor used in the simulation is constructed via RBF interpolation as in [15].

The data set takes into account *Vinho Verde* wine (from the Minho region of Portugal; see http://www.vinhoverde.pt/) with a total of 1599 red samples. Based on 11 inputs each wine is classified on a quality range that varies from 0 to 10. We briefly describe below the inputs and we refer to [9, 12, 32] for further details.

VA. Volatile acidity (tartaric acid - g /dm^3): it is the amount of acetic acid in wine. It is a relevant physicochemical parameter and at too high levels it can lead to an unpleasant vinegar taste. As a consequence, it is monitored during the winemaking process.

FA. Fixed acidity (acetic acid - g / dm^3): it is represented by the acids that do not evaporate readily. It is defined as the difference among the total acidity and the volatile acidity. The former, in must or wine, takes into account all types of acids.

CA. Citric acid (g / dm^3): it appears in small quantities and adds freshness and flavor to wines.

RS. Residual sugar (g/dm^3): it measures the amount of sugar solids remaining after fermentation. Dry wines are characterized by low levels of residual sugar, while if the latter is greater than 45 g/dm^3, then the wine is usually referred to as sweet.

CH. Chlorides (sodium chloride - g/dm^3): such measure corresponds to the amount of salt in the wine.

FD. Free sulfur dioxide (mg / dm^3): it prevents microbial growth and the oxidation of wine.

TD. Total sulfur dioxide (mg / dm^3): it appears in small quantities and becomes evident in the nose only for high concentrations of free sulfur dioxide.

DE. Density (g / cm^3): it is close to that of water depending on the alcohol and sugar concentration.

PH. pH: it describes how acidic or basic a wine is. Its scale is from 0 to 14. Since it influences microbiological stability and affects the equilibrium of tartrate salts, it is largely responsible for the flavor balance [9].

PS. Potassium sulphate (g / dm^3); it is a wine additive which might affect the sulfur dioxide gas levels.

AL. Alcohol: the percentage of alcohol content of the wine. It is used to express the wine's strength.

The output variable, i.e. the Wine Quality (WQ), based on sensory data, is a score between 0 and 10. Thus, to predict the final score we need to approximate the 11-variate function f that classifies the wine and depends on the above parameters, i.e.

$$\mathrm{WQ} = f(\mathrm{VA},\mathrm{FA},\mathrm{CA},\mathrm{RS},\mathrm{CH},\mathrm{FD},\mathrm{TD},\mathrm{DE},\mathrm{PH},\mathrm{PS},\mathrm{AL}).$$

About the 10% of the data set, specifically $s = 160$ instances, is used for testing the methods.

The results are reported in Table 3. We can note that, the simple interpolation via RBF suffers from the curse of dimensionality. This stresses the fact that, even if RBFs can be applied in any dimensions, the use of tensor decomposition tools becomes essential.

6 Concluding remarks

In this paper we proposed a technique which turns out to be extremely useful in approximating functions in high dimensions. The main advantage consists in the fact that we reduce the large scale problem in interpolating eigenfunctions, which are univariate functions. Both theoretical and numerical evidence confirm that the investigated tools turn out to be robust when approximating multivariate functions. Work in progress consists in coupling local schemes, as the PU method, with the TD tools and investigating applications to the solution of PDEs via collocation algorithms. Moreover, because of the flexibility of the current scheme with respect to the basis functions, extensions to multivariate polynomial interpolation without resampling [5, 16] need further investigations.

Acknowledgements

This research has been accomplished within Rete ITaliana di Approssimazione (RITA), partially funded by GNCS-INδAM and by the project "Artificial Intelligence for Solar Flares" (UNIGE). We sincerely thank the reviewers for helping us to significantly improve the manuscript. Special thanks are also due to Prof. Stefano De Marchi who taught me to appreciate wines in numerous workshops in Canazei.

References

[1] M. Azaïez, F. Ben Belgacem, *Karhunen-Loève's truncation error for bivariate functions*, Comput. Methods in Appl. Mech. Eng. **290**, (2015), pp. 57–72.

[2] M. Azaïez, F. Ben Belgacem, T. Chacón Rebollo, *Recursive POD expansion for reaction-diffusion equation*, Adv. Model. and Simul. in Eng. Sci. (2016) 3:3. DOI 10.1186/s40323-016-0060-1.

[3] M. Azaïez, T. Chácon Rebollo, E. Perracchione, J. M. Vega, *Recursive POD expansion for advection-diffusion-reaction equation*, Comm. Comput. Physics **24** (2018), pp. 1556–1578.

[4] M. Balajewicz, *A new approach to model order reduction of the Navier-Stokes Equations*, PhD thesis, Duke University, Durham, 2012.

[5] J.P. Berrut, S. De Marchi, G. Elefante, F. Marchetti, *Treating the Gibbs phenomenon in barycentric rational interpolation via the S-Gibbs algorithm*, Appl. Math. Letters **103** (2020), 106196.

[6] G. Berkooz, P. Holmes, J.L. Lumley, *The proper orthogonal decomposition in the analysis of turbulent flows*, Annu. Rev. Fluid Mech. **25** (1993), pp. 539–575.

[7] M.D. Buhmann, *Radial Basis Functions: Theory and Implementation*, Cambridge Monogr. Appl. Comput. Math., vol. 12, Cambridge Univ. Press, Cambridge, 2003.

[8] M. Buhmann, S. De Marchi, E. Perracchione, *Analysis of a new class of rational RBF expansions*, to appear on IMA J. Numer. Anal. 2019; https://doi.org/10.1093/imanum/drz015.

[9] R.B. Boulton, V.L. Singleton, L.F. Bisson, R.E. Kunkee, *Principles and practices of winemaking*, New York, Chapman & Hall, 1996.

[10] R. Cavoretto, *A numerical algorithm for multidimensional modeling of scattered data points*, Comput. Appl. Math. **34** (2015), pp. 65–80.

[11] R. Cavoretto, A. De Rossi, *A trivariate interpolation algorithm using a cube-partition searching procedure*, SIAM J. Sci. Comput. **37** (2015), pp. A1891–A1908.

[12] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, J. Reis, *Modeling wine preferences by data mining from physicochemical properties*, Decis. Support Syst. **47** (2009), pp. 547–553.

[13] L. De Lathawer, B. De Moor, J. Vandewalle, *On the best rank-1 and rank-*$(R_1, R_2, \ldots, R_N)$ *approximation of higher-order tensors*, SIAM J. Matrix. Anal. Appl. **21** (2000), pp. 1324–1342.

[14] S. De Marchi, W. Erb, F. Marchetti, *Spectral filtering for the reduction of the Gibbs phenomenon for polynomial approximation methods on Lissajous curves with applications in MPI*, Dolomites Res. Notes Approx, **10** (2017), pp. 128–137.

[15] S. De Marchi, F. Marchetti, E. Perracchione, Jumping with Variably Scaled Discontinuous Kernels (VSDKs), to appear on BIT, https://doi.org/10.1007/s10543-019-00786-z.

[16] S. De Marchi, F. Marchetti, E. Perracchione, D. Poggiali, *Polynomial interpolation via mapped bases without resampling*, J. Comput. Appl. Math. **364**, 2020.

[17] S. De Marchi, A. Martínez, E. Perracchione, *Fast and stable rational RBF-based partition of unity interpolation*, J. Comput. Appl. Math. **349** (2019), pp. 331–343.

[18] S. De Marchi, G. Santin, *Fast computation of orthonormal basis for RBF spaces through Krylov space methods*, BIT **55** (2015), pp. 949–966.

[19] B. Diederichs, A. Iske, *Improved estimates for condition numbers of radial basis function interpolation matrices*, J. Approx. Theory **238** (2019), pp. 38–51.

[20] G.E. Fasshauer, *Meshfree Approximations Methods with* Matlab, World Scientific, Singapore, 2007.

[21] G.E. Fasshauer, M.J. McCourt, *Kernel-based Approximation Methods Using* Matlab, World Scientific, Singapore, 2015.

[22] R.A. Horn, F. Zhang, *Bounds on the spectral radius of a Hadamard product of nonnegative or positive semidefinite matrices*, Electron. J. Linear Algebra **20** (2010), pp. 90–94.

[23] N. El Karoui, *The spectrum of kernel random matrices*, Ann. Statist. **38** (2010), pp. 1–50.

[24] S. Jakobsson, B. Andersson, F. Edelvik, *Rational radial basis function interpolation with applications to antenna design*, J. Comput. Appl. Math. **233** (2009), pp. 889–904.

[25] E. Larsson, V. Shcherbakov, A. Heryudono, *A least squares radial basis function partition of unity method for solving PDEs*, SIAM J. Sci. Comput. **39** (2017), pp. A2538–A2563.

[26] M.M. Loève, *Probability Theory*, Van Nostrand, Princeton, NJ, 1988.

[27] K.N. Majindar, *On a factorization of positive definite matrices*, Canadian Math. Bull. **6** (1963), pp. 405–407.

[28] I. Martini, B. Haasdonk, G. Rozza, *Certified reduced basis approximation for the coupling of viscous and inviscid parametrized flow models*, J. Sci. Comput. **74** (2018), pp. 197–219.

[29] J. Mercer, *Functions of positive and negative type and their connection with the theory of integral equations*, Phil. Trans. Royal Society, **209** (1909), pp. 415–446.

[30] E. Perracchione, *Rational RBF-based partition of unity method for efficiently and accurately approximating 3D objects*, Comput. Appl. Math. (2018), **37**, pp. 4633–4648

[31] S.A. Sarra, Y. Bay, *A rational radial basis function method for accurately resolving discontinuities and steep gradients*, Appl. Num. Math. **130** (2018), pp. 131–142.

[32] P. Ribéreau-Gayon, Y. Glories, A. Maujean, D. Dubourdieu, *Handbook of Enology, Volume 2, The Chemistry of Wine Stabilization and Treatments, 2nd Edition*, John Wiley & Sons Ltd, England, 2006.

[33] A. Safdari-Vaighani, E. Larsson, A. Heryudono, *Radial basis function methods for the Rosenau equation and other higher order PDEs*, J. Sci. Comp. **75** (2018), pp. 1555–1580.

[34] J. Schur, *Bemerkungen zur Theorie der beschranktcn Bilinearformen mit unendlich vielcn Veranderlichen*, J. Reine Angew. Math. **140** (1911), pp. 1–28.

[35] A. SCHMIDT, B. HAASDONK, *Reduced basis approximation of large scale parametric algebraic Riccati Equations*, ESAIM: COCV **24** (2018), pp. 129–151.

[36] H. WENDLAND, *Scattered Data Approximation*, Cambridge Monogr. Appl. Comput. Math., vol. 17, Cambridge Univ. Press, Cambridge, 2005.

[37] H. WENDLAND, *Fast evaluation of radial basis functions: Methods based on partition of unity*, in: C.K. Chui et al. (Eds.), Approximation Theory X: Wavelets, Splines, and Applications, Vanderbilt Univ. Press, Nashville, 2002, pp. 473–483.