Multivariate approximation at fake nodes

S. De Marchi[*†], F. Marchetti[**], E. Perracchione[*], D. Poggiali[†]

[*] *Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova, Italy;*
[**] *Dipartimento di Salute della Donna e del Bambino, Università di Padova, Italy;*
[†] *PNC - Padova Neuroscience Center, Università di Padova, Italy*

Abstract

The main goal of the present paper is to extend the use of the so-called *mapped bases without resampling* to *any* basis and dimension. Indeed, it has been previously investigated only for univariate (rational) polynomial interpolation. The concept of mapped bases has been widely studied, but all the proposed methods show convergence provided that the function is resampled at the mapped nodes. In applications, this is often physically unfeasible. Thus, we propose an effective method for interpolating via mapped bases the given function values for multivariate interpolation schemes. We might refer to the method as *fake nodes* approach. Numerical experiments and theoretical studies are devoted to show the robustness of the proposed scheme.

Keywords: Kernel interpolation, polynomial interpolation, Gibbs phenomenon, Runge phenomenon, mapped bases.
2010 MSC: 65D05, 41A05, 65D15.

1. Introduction

The multivariate scattered data approximation has a huge variety of applications and, in this sense, is one of the most attractive topics in numerical analysis. Many methods, such as multivariate splines, meshfree or meshless approaches and finite elements [9, 12, 16, 37, 38], have already been proven to be effective numerical tools and, compared to polynomial bases (see e.g. [15]), are easier to implement in higher dimensions. This is mainly due to the fact that the polynomial basis is not data-dependent. Therefore, identifying unisolvent data sets for multivariate polynomial interpolation is a fundamental step, refer e.g. to [7, 22, 23].

In what follows we investigate how the so-called *fake nodes* approach, first introduced for univariate and for rational polynomial interpolation in [5, 21], can

Email addresses: demarchi@math.unipd.it (S. De Marchi[*†]), francesco.marchetti.1@phd.unipd.it (F. Marchetti[**]), emma.perracchione@math.unipd.it (E. Perracchione[*]), poggiali.davide@gmail.com (D. Poggiali[†])

be successfully used for multivariate interpolation or approximation schemes. It is based on mapping points (see [1, 4, 30]), i.e. bases for data-dependent methods, without any need of resampling the usually unknown function and taking somehow into account the behaviour of the latter. All the above mentioned numerical schemes are affected by the Gibbs phenomenon (see e.g. [2, 27, 35]) that can be mitigated via the approach investigated in what follows. We also show for kernel-based methods that the mapped bases scheme is similar to the so-called Variably Scaled Discontinuous Kernels (VSDKs); refer e.g. to [8, 19, 20, 34].

For polynomial approximation, aside the Gibbs phenomenon, we also face two other important aspects. Indeed, finding unisolvent sets and mitigating the Runge [36] phenomenon is not trivial. To accommodate both issues we propose a mapped interpolation on multidimensional Chebyshev grids and on the so-called Padua points; refer to [7, 22, 23, 26]. Theoretical studies, with a particular focus on the *Lebesgue constant* (see e.g. [11, 17, 32]), and various numerical experiments are devoted to show the efficacy, the easy implementation and hence the applicability of the fake nodes approach to many bases.

The guidelines of the paper are as follows. In Section 2 we review the basics of multivariate interpolation and approximation schemes. The fake nodes and their theoretical properties are presented in Section 3. Sections 4 and 5 focus on two specific maps and on the expansion of the approximant for kernel and polynomial bases. Numerical tests are presented in Section 6. The last section deals with conclusions and work in progress.

2. Preliminaries

In many fields of application, the multivariate scattered data interpolation problem defined below arises.

Problem 2.1 (Multivariate scattered data interpolation) *Given a set of scattered data* $X_N = \{\boldsymbol{x}_i,\ i = 1, \dots, N\} \subseteq \Omega$*, with* $\Omega \subset \mathbb{R}^d$ *and the associated function values* $F_N = \{f(\boldsymbol{x}_i),\ i = 1, \dots, N\}$*, which are sampled from a function* f*, find a function* P_f *so that*

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

To review the basics of approximation theory, we mainly refer to the books [16, 25, 37].

A common approach (e.g. for polynomials, RBFs and splines) for the scattered data interpolation problem is to assume that

$$P_f \in H_N \coloneqq \operatorname{span}\{B_1, \dots, B_N\},$$

where $B_i : \Omega \longrightarrow \mathbb{R}$, $i = 1, \dots, N$, are certain *basis functions*. Starting from them, we can also recover the so-called *cardinal* functions $u_i \in H_N$ so that $u_i(\boldsymbol{x}_j) = \delta_{ij}$, $i, j = 1, \dots, N$, by solving

$$\mathsf{R}\boldsymbol{u} = \boldsymbol{b},$$

where $\mathsf{R}_{ij} = B_i(\boldsymbol{x}_j)$, $i,j = 1,\ldots,N$, $\boldsymbol{u} = (u_1(\boldsymbol{x}),\ldots,u_N(\boldsymbol{x}))^\intercal$ and the vector on the right hand side is given by $\boldsymbol{b} = (B_1(\boldsymbol{x}),\ldots,B_N(\boldsymbol{x}))^\intercal$.

Given the cardinal functions, we can write the interpolant in its cardinal (also called *lagrangian*) form as

$$P_f(\boldsymbol{x}) = \sum_{i=1}^{N} f(\boldsymbol{x}_i)u_i(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega.$$

The cardinal form allows to introduce the so-called *Lebesgue constant*, which is the key ingredient for stability analysis. It is given by ([6, 11])

$$\Lambda(\Omega) = \sup_{\boldsymbol{x}\in\Omega} \sum_{i=1}^{N} |u_i(\boldsymbol{x})|.$$

Moreover, to solve the scattered data problem, it might be convenient to define the interpolant as

$$P_f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i B_i(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega.$$

Indeed, for finding the coefficients α_i, $i = 1,\ldots,N$, we reduce to solving a linear system that assumes the form

$$\mathsf{A}\boldsymbol{\alpha} = \boldsymbol{f}, \tag{1}$$

where $\mathsf{A}_{ij} = B_i(\boldsymbol{x}_j)$, $i,j = 1,\ldots,N$, $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_N)^\intercal$ and the vector of function values is given by $\boldsymbol{f} = (f(\boldsymbol{x}_1),\ldots,f(\boldsymbol{x}_N))^\intercal$.

Remark 2.1 *The problem defined in* (1) *is well-posed if and only if the matrix* A *is non-singular. For instance, in the univariate setting, we can interpolate* N *distinct data with a polynomial of degree* $N-1$*. For the multivariate setting, the existence and uniqueness of the solution of a scattered data interpolation problem is not always satisfied. This fact is related to the so-called* Haar systems *and* unisolvent sets*. For a detailed analysis of such concepts, see e.g. [15, 37].*

Definition 2.1 (Haar system) *The finite-dimensional linear space* $H_N \subseteq C(\Omega)$*, with basis* $\{B_i\}_{i=1}^N$*, is a Haar space on* Ω *if*

$$\det\mathsf{A} \neq 0,$$

for any set of distinct data points $X_N = \{\boldsymbol{x}_i, i = 1,\ldots,N\} \subseteq \Omega$*. The set* $\{B_i\}_{i=1}^N$ *is called a Haar system.*

In the multivariate case, there exist no non-trivial Haar spaces. This is a consequence of the famed Haar-Mairhuber-Curtis theorem [14, 28, 31].

Theorem 2.1 (Haar-Mairhuber-Curtis) *Suppose that* $\Omega \subseteq \mathbb{R}^d$*,* $d \geq 2$*, contains an interior point. Then there exist no Haar spaces of continuous functions except for trivial ones, i.e. spaces spanned by a single function.*

According to Definition 2.1, saying that $\{B_i\}_{i=1}^N$ is a Haar system on Ω is equivalent to state that such a basis is unisolvent on Ω. On the other hand, the notion of unisolvency for sets of points is different.

Definition 2.2 (Unisolvent set) *A finite set of point $X_N = \{\boldsymbol{x}_i, i = 1, \dots, N\} \subseteq \Omega$ is unisolvent for H_N if all the elements of H_N are completely determined by their values at X_N.*

While we do not dispose of non-trivial unisolvent bases in the multivariate setting, it is possible to consider multidimensional unisolvent sets of points. For example, the Padua points are unisolvent for bivariate polynomial interpolation of total degree. Moreover, any set of distinct nodes are unisolvent with respect to a data-dependent kernel basis.

Therefore, when possible, we consider sets of nodes that are unisolvent with respect to the chosen basis $\{B_i\}_{i=1}^N$, since this guarantees the uniqueness of the interpolant in H_N.

Remark 2.2 (Least squares) *If the set of nodes is not unisolvent, we relax the interpolation conditions and thus approximate the function f in the least squares sense, i.e. f is approximated in the space $H_m \coloneqq span\{B_1, \dots, B_m\}$, with $m < N$. Unless otherwise noted, in what follows we focus on the interpolation problem.*

Independently of the selected basis, we might numerically observe non-physics oscillations in the resulting interpolant. They are due to both Gibbs and Runge phenomena. Indeed, they appear with different characteristics depending on the basis functions, but they are common to all interpolation methods, e.g. polynomial expansions and Radial Basis Functions (RBFs) approximants. Many efforts are devoted to mitigate the effects of such phenomena. For this purpose, many techniques are based on resampling the function at a different set of nodes. Since this might not be possible in many applications, the relevance of investigating the *fake nodes* tool becomes evident.

3. Fake nodes as a general approach

Let us consider an injective map $S : \Omega \longrightarrow \mathbb{R}^d$. The idea is to construct an interpolant

$$R_f \in H_N^S \coloneqq \text{span}\{B_1^S, \dots, B_N^S\},$$

so that for $\boldsymbol{x} \in \Omega$

$$R_f(\boldsymbol{x}) = \sum_{i=1}^N \alpha_i^S B_i^S(\boldsymbol{x}) = \sum_{i=1}^N \alpha_i^S B_i(S(\boldsymbol{x})) = P_g(S(\boldsymbol{x})), \tag{2}$$

and the function g is such that $g_{|S(X_N)} = f_{|X_N}$; see [5, 21]. Thus, constructing the interpolant R_f considering the mapped basis is equivalent to build a classical interpolant $P_g \in H_N$ at the fake nodes $S(X_N)$.

To define the interpolant R_f, provided that we have a unisolvent set of points for the given basis, we solve

$$\mathsf{A}^S \boldsymbol{\alpha}^S = \boldsymbol{f}, \tag{3}$$

where $\boldsymbol{\alpha}^S = (\alpha_1^S, \ldots, \alpha_N^S)^\intercal$, $\boldsymbol{f} = (f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N))^\intercal$, and

$$\mathsf{A}^S = \begin{pmatrix} B_1^S(\boldsymbol{x}_1) & \ldots & B_1^S(\boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ B_N^S(\boldsymbol{x}_1) & \ldots & B_N^S(\boldsymbol{x}_N) \end{pmatrix}.$$

If the considered set of nodes is not unisolvent, we relax the interpolation conditions looking for a least square approximant on $H_m^S \coloneqq \operatorname{span}\{B_1^S, \ldots, B_m^S\}$, with $m < N$.

Concerning the cardinal form of the interpolant we have the following result.

Proposition 3.1 (Cardinal basis equivalence) *Let $X_N = \{\boldsymbol{x}_i,\ i = 1, \ldots, N\} \subseteq \Omega$, with $\Omega \subset \mathbb{R}^d$ be a set of distinct scattered data and let $u_i \in H_N$, $i = 1, \ldots, N$ be the associated cardinal functions. Let $S : \Omega \longrightarrow \mathbb{R}^d$ be an injective map. The functions $\boldsymbol{u} = (u_1, \ldots, u_N)^\intercal$ are cardinal on $S(\Omega)$ for the fake nodes $S(X_N)$ if and only if the mapped functions $\boldsymbol{u}^S = (u_1 \circ S, \ldots, u_N \circ S)^\intercal$ are cardinal for the original set of nodes X_N.*

Proof: Because of the definition of the interpolant, we have that

$$u_i^S(\boldsymbol{x}_j) = u_i(S(\boldsymbol{x}_j)) = \delta_{i,j} \quad i, j = 1, \ldots, N.$$

∎

By virtue of the above proposition, we can also write the interpolant at the fake nodes in *cardinal* form as

$$R_f^S(\boldsymbol{x}) = \sum_{i=1}^{N} f(\boldsymbol{x}_i) u_i^S(\boldsymbol{x}) = \sum_{i=1}^{N} f(\boldsymbol{x}_i) u_i(S(\boldsymbol{x})), \quad \boldsymbol{x} \in \Omega. \tag{4}$$

We now want to the cardinal form of the interpolant based on the fake nodes. To this aim, we recall the following result (see [29], [10, Theorem 3, p. 3] and [13, Exercise 15, p.64]).

Theorem 3.2 *Given a (finite) unisolvent set of nodes $X_N \subseteq \Omega$ for the space $H_N = \operatorname{span}\{B_1, \ldots, B_N\}$ and the associated function values F_N, the determinant form of the interpolant P_f for $\boldsymbol{x} \in \Omega$ is given by*

$$P_f(\boldsymbol{x}) = -\frac{1}{\det(\mathsf{A})} \det \begin{pmatrix} 0 & B_1(\boldsymbol{x}) & \ldots & B_N(\boldsymbol{x}) \\ f_1(\boldsymbol{x}) & B_1(\boldsymbol{x}_1) & \ldots & B_N(\boldsymbol{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ f_N(\boldsymbol{x}) & B_1(\boldsymbol{x}_N) & \ldots & B_N(\boldsymbol{x}_N) \end{pmatrix}.$$

Proof: Following the steps of a similar Theorem in [29], we define the function

$$q(\boldsymbol{x}) = \det \begin{pmatrix} 0 & B_1(\boldsymbol{x}) & \dots & B_N(\boldsymbol{x}) \\ f_1(\boldsymbol{x}) & B_1(\boldsymbol{x}_1) & \dots & B_N(\boldsymbol{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ f_N(\boldsymbol{x}) & B_1(\boldsymbol{x}_N) & \dots & B_N(\boldsymbol{x}_N) \end{pmatrix} \in H_N.$$

The evaluation of such function in a generic node $\boldsymbol{x}_i,\ i = 1, \dots, N$ can be computed by subtracting the i-th row to the first

$$q(\boldsymbol{x}_i) = \det \begin{pmatrix} -f_i & 0 & \dots & 0 \\ f_1(\boldsymbol{x}) & B_1(\boldsymbol{x}_1) & \dots & B_N(\boldsymbol{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ f_N(\boldsymbol{x}) & B_1(\boldsymbol{x}_N) & \dots & B_N(\boldsymbol{x}_N) \end{pmatrix} = -f_i \det(\mathsf{A}).$$

If follows that the function $Q(\boldsymbol{x}) = -\dfrac{1}{\det(\mathsf{A})} q(\boldsymbol{x})$ belongs as well to H_N and is such that $Q(\boldsymbol{x}_i) = f_i\ \forall i = 1, \dots, N$. The thesis follows from the uniqueness of the interpolant.

∎

By virtue of the above theorem, for the computation of the interpolant on the fake nodes in its cardinal form, we have the following result.

Corollary 3.2.1 (Determinant form) *Let $X_N \subseteq \Omega$ be a unisolvent set of nodes for the space H_N and let F_N be the associated function values. Let $S : \Omega \longrightarrow \mathbb{R}^d$ be an injective map and $u_i^S \in H_N^S$, $i = 1, \dots, N$, be the cardinal functions. For $\boldsymbol{x} \in \Omega$, the interpolant R_f^S can be computed as*

$$R_f^S(\boldsymbol{x}) = -\det \mathsf{U}_N, \tag{5}$$

with

$$\mathsf{U}_N = \begin{pmatrix} 0 & u_1^S(\boldsymbol{x}) & \dots & u_N^S(\boldsymbol{x}) \\ f_1(\boldsymbol{x}) & & & \\ \vdots & & \mathsf{I}_N & \\ f_N(\boldsymbol{x}) & & & \end{pmatrix},$$

and where I_N denotes the $N \times N$ identity matrix.

Proof: The proof directly follows from Theorem 3.2. Alternatively, it can be shown by induction. For $N = 1$ and $\boldsymbol{x} \in \Omega$, we have that

$$R_f^S(\boldsymbol{x}) = -\det \mathsf{U}_1 = -\det \begin{pmatrix} 0 & u_1^S(\boldsymbol{x}) \\ f_1(\boldsymbol{x}) & 1 \end{pmatrix} = f_1(\boldsymbol{x}) u_1^S(\boldsymbol{x}).$$

We suppose that the assertion holds true for $N - 1$. Then, computing the determinant for the last row we have that

$$R_f^S(\boldsymbol{x}) = -\det \mathsf{U}_N = -\left[(-1)^{N+2}(-1)^{N+1} f_N(\boldsymbol{x}) u_N^S(\boldsymbol{x}) \det \mathsf{I}_N + (-1)^{2N+2} \det \mathsf{U}_{N-1}\right].$$

By induction, we obtain that

$$\begin{aligned} R_f^S(\boldsymbol{x}) &= -\left(f_N(\boldsymbol{x})u_N^S(\boldsymbol{x}) - \det \mathsf{U}_{N-1}\right) \\ &= \left(f_N(\boldsymbol{x})u_N^S(\boldsymbol{x}) + f_{N-1}(\boldsymbol{x})u_{N-1}^S(\boldsymbol{x}) + \ldots + f_1(\boldsymbol{x})u_1^S(\boldsymbol{x})\right), \end{aligned}$$

which proves the equivalence between (4) and (5). ∎

Concerning the Lebesgue function associated to R_f^S, we have the following result.

Proposition 3.3 (Equivalence of the Lebesgue constant) *Let $S : \Omega \longrightarrow \mathbb{R}^d$ be an injective map. Given a unisolvent set of nodes $X_N \subseteq \Omega$ for the space H_N, and the associated cardinal functions $u_i^S \in H_N^S$, $i = 1, \ldots, N$, the Lebesgue constant associated to the mapped nodes $\Lambda^S(\Omega)$ is so that*

$$\Lambda^S(\Omega) = \Lambda(S(\Omega)).$$

Proof: By definition of $\Lambda^S(\Omega)$, for $\boldsymbol{x} \in \Omega$ we trivially have that:

$$\Lambda^S(\Omega) = \sup_{\boldsymbol{x}\in\Omega} \sum_{i=1}^{N} |u_i^S(\boldsymbol{x})| = \sup_{\boldsymbol{x}\in\Omega} \sum_{i=1}^{N} |u_i(S(\boldsymbol{x}))| = \sup_{\boldsymbol{y}\in S(\Omega)} \sum_{i=1}^{N} |u_i(\boldsymbol{y})| = \Lambda(S(\Omega)).$$

∎

This proposition states that the interpolation on the mapped basis H_N^S inherits the Lebesgue constant of the fake nodes $S(X_N)$ over the 'standard' basis H_N. The Lebesgue constant is extremely helpful for studying the stability of the interpolant. To this aim, let us now consider an interpolant of perturbed data $\tilde{f}(\boldsymbol{x}_i)$ sampled at $\boldsymbol{x}_i$, $i = 1, \ldots, N$. For instance, the perturbation might be due to due to measurements errors. In that case, the uncertainty on the data propagates as follows.

Proposition 3.4 (Stability) *Let $S : \Omega \longrightarrow \mathbb{R}^d$ be an injective map and $X_N \subseteq \Omega$ be a unisolvent set of nodes for the space H_N. Let $f(\boldsymbol{x}_i)$ be the associated function values and $\tilde{f}(\boldsymbol{x}_i)$, $i = 1, \ldots, N$, be the perturbed data. Let R_f^S and $R_{\tilde{f}}^S$ be the interpolant of the function values $f(\boldsymbol{x}_i)$ and $\tilde{f}(\boldsymbol{x}_i)$, $i = 1, \ldots, N$, respectively. Then,*

$$||R_f^S - R_{\tilde{f}}^S||_{\infty,\Omega} \leq \Lambda^S(\Omega)\, \|f - \tilde{f}\|_{\infty,X_N}.$$

Proof: Taking into account that $g_{|S(X_N)} = f_{|X_N}$ and thus also $\tilde{g}_{|S(X_N)} = \tilde{f}_{|X_N}$,

we deduce that

$$||R_f^S - R_{\tilde{f}}^S||_{\infty,\Omega} = ||P_g - P_{\tilde{g}}||_{\infty,S(\Omega)} = \sup_{x\in S(\Omega)} \left| \sum_{i=1}^{N} (g_i(\boldsymbol{x}_i) - \tilde{g}_i(\boldsymbol{x}_i))\ u_i(\boldsymbol{x}) \right| =$$

$$= \sup_{x\in\Omega} \left| \sum_{i=1}^{N} (g_i(S(\boldsymbol{x}_i)) - \tilde{g}_i(S(\boldsymbol{x}_i)))\ u_i(S(\boldsymbol{x})) \right| \leq$$

$$\leq \sup_{x\in\Omega} \sum_{i=1}^{N} |u_i(S(\boldsymbol{x}))|\ \ |g_i(S(\boldsymbol{x_i})) - \tilde{g}_i(S(\boldsymbol{x_i}))| \leq$$

$$\leq \sup_{x\in\Omega} \sum_{i=1}^{N} |u_i(S(\boldsymbol{x}))|\ \max_{i=1,\dots,N} |g_i(S(\boldsymbol{x_i})) - \tilde{g}_i(S(\boldsymbol{x_i}))| =$$

$$= \Lambda(S(\Omega))\ \max_{i=1,\dots,N} \left| f(\boldsymbol{x}_i) - \tilde{f}_i(\boldsymbol{x}_i) \right|$$

$$= \Lambda^S(\Omega)\ \|f - \tilde{f}\|_{\infty,X_N}.$$

132 ∎

Proposition 3.5 (Error bound inheritance) *Let $S : \Omega \longrightarrow \mathbb{R}^d$ be an injective map. Let $X_N \subseteq \Omega$ be a unisolvent set of nodes for the space H_N and $f(\boldsymbol{x}_i)$ be the associated function values $i = 1, \dots, N$. Let R_f^S be the interpolant of the function values $f(\boldsymbol{x}_i)$, $i = 1, \dots, N$,. Then, R_f^S inherits the error bounds of the original bases in any given norm, i.e.*

$$||R_f^S - f||_\Omega = ||P_g - g||_{S(\Omega)},$$

133 *where g is so that $\tilde{g}_{|S(X_N)} = \tilde{f}_{|X_N}$.*

Proof: Given that $R_f^S = P_g \circ S$ as in (2) and choosing the function g such that $g \circ S = f$ on Ω (this function exists since the map S is injective), we get

$$||R_f^S - f||_\Omega = ||P_g \circ S - g \circ S||_\Omega = ||P_g - g||_{S(\Omega)}.$$

134 ∎

135 In what follows, we focus on two specific maps and we extend the S-Runge
136 and S-Gibbs algorithms, introduced in [21], to the multidimensional setting.

137 4. S-Gibbs

138 We face the problem of interpolating discontinuous functions. In particular,
139 we consider the following general setting.

140 **Assumption 4.1 (For S-Gibbs)** *We suppose that Ω is the union of p pairwise*
141 *disjoint sets Ω_k, $k = 1, \dots, p$, and that f is piecewise continuous. In particular,*
142 *the discontinuities of f appear only at the boundaries of the subsets $\partial\Omega_k$, $k =$*
143 *$1, \dots, p$. As an example, please refer to Figure 1 (left).*

Under such assumptions, we select the function S as

$$S(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{a}_i, \tag{6}$$

for $\boldsymbol{x} \in \Omega_i$ and $\boldsymbol{a}_i = a_i \text{ diag}(\mathsf{I}_d)$, $a_i \in \mathbb{R}$, $i = 1, \dots, p$. The latter should be chosen so that we obtain p disjoint sets, according to the discontinuities of f. For an example refer to Figure 1 (right).

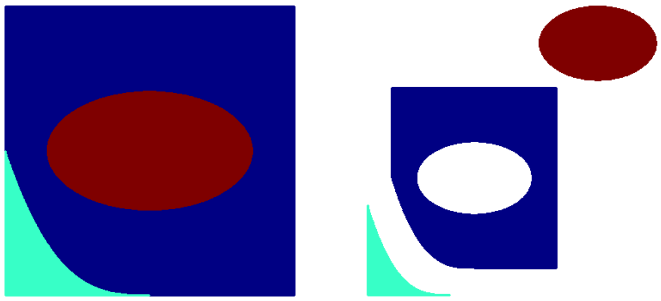

Figure 1: Left: example of possible domain Ω under Assumption 4.1. Right: the mapped domain via the S-Gibbs map S.

Before going into details with the numerical experiments, we focus on kernel bases and more specifically on the so-called Variably Scaled Discontinuous Kernels (VSDKs).

4.1. A note on kernel bases

The method described in the previous section works for any basis. In particular, for kernels (refer e.g. to [12]), it shows strong similarities to the so-called VSDKs [8, 19, 20]. For kernel-based approximants, we take $P_f \in \text{span}\{\kappa(\cdot, \boldsymbol{x}_1), \dots, \kappa(\cdot, \boldsymbol{x}_N)\}$, where $\kappa : \Omega \times \Omega \longrightarrow \mathbb{R}$ is a strictly positive definite radial kernel. Since it is radial, we can then associate a univariate function $\phi : [0, \infty) \longrightarrow \mathbb{R}$ so that:

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \phi(r), \quad \text{where} \quad r = \|\boldsymbol{x} - \boldsymbol{y}\|_2 .$$

We also point out that usually the distance r is rescaled via the so-called *shape parameter* $\gamma \in \mathbb{R}, \gamma > 0$, i.e. we consider $\phi(\gamma r)$ as basis function.

Then, to determine the kernel interpolant, we reduce to solving a system of the form (1), with $\mathsf{A} = \phi(\mathsf{D})$, where the so-called distance matrix D is given by

$$\mathsf{D}_{ij} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2.$$

Note that, in case we use the S-Gibbs map (6), then the kernel matrix is given by $\mathsf{A}^S = \phi(\mathsf{D}^S)$, where the entries of D^S are defined as

$$\mathsf{D}^S_{ij} = ||S(\boldsymbol{x}_i) - S(\boldsymbol{x}_j)||_2.$$

To mitigate the Gibbs phenomenon, the VSDKs have been introduced; refer to [8, 19, 20]. Precisely, given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ and a function $\psi : \mathbb{R}^d \longrightarrow \mathbb{R}$, the idea is to consider the kernel

$$\kappa^{\psi}(\boldsymbol{x}, \boldsymbol{y}) \coloneqq \kappa\left((\boldsymbol{x}, \psi(\boldsymbol{x})), (\boldsymbol{y}, \psi(\boldsymbol{y}))\right),$$

where κ is a kernel on $\mathbb{R}^{d+1}$. Therefore, we produce an *augmented* set of nodes $\tilde{X}_N = \{\tilde{\boldsymbol{x}}_i = (\boldsymbol{x}_i, \psi(\boldsymbol{x}_i)),\ i = 1, \dots, N\} \subseteq \Omega$ on $\mathbb{R}^{d+1}$. Then, on $\mathbb{R}^{d+1}$ we use standard kernels for the interpolation. In other words, we solve a system of the form (1) whose kernel matrix is defined as $\mathsf{A}^{\psi} = \phi(\mathsf{D}^{\psi})$, where

$$\mathsf{D}^{\psi}_{ij} = ||\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j||_2.$$

Under the Assumption 4.1, in [20] we select the function ψ piecewise constant, i.e. as $\psi(\boldsymbol{x}) = b_i$, $b_i \in \mathbb{R}$, for $\boldsymbol{x} \in \Omega_i$. Being the RBF interpolation methods dependent on the distance among the nodes, we observe that the VSDKs are *similar* to the fake nodes approach, i.e. $\mathsf{A}^{\psi} \approx \mathsf{A}^{S}$. Indeed, we observe that both VSDKs and fake nodes approach preserve the distances between points that lie in the same subdomain $\Omega_k \subset \Omega$, while they enlarge the distances between points lying in different subregions in Ω. To graphically show this, we report in Figure 2 the effect of applying the fake nodes and the VSKs on 12 nodes under the hypothesis of having a discontinuity in $x = 0.5$.

After studying the S-Runge algorithm, we show the efficacy of our method via several numerical experiments.

5. S-Runge

For multivariate interpolation, we are not able to give any general receipt. Indeed, different node sets, depending on the basis functions, could be considered. In this section, we focus on polynomial interpolation, since it is known to be heavily affected by the Runge phenomenon in some cases. The main idea consists in mapping points via a function S such that $\Lambda^S(\Omega) \leq \Lambda(\Omega)$. This can be achieved thanks to the fact that we inherit the Lebesgue constant on the mapped nodes.

In what follows, we present two different methods to deal with the Runge phenomenon in a multidimensional setting. While the former is built upon recursive one-dimensional reconstructions and it can be extended to any dimensions, with the latter we restrict to the two-dimensional case and we take advantage of the *optimal* properties of the so-called *Padua points*.

5.1. The multidimensional lines approach

In case of univariate polynomial interpolation, we know that the Chebyshev-Lobatto (CL) nodes represent an optimal choice, meaning that the Lebesgue constant grows logarithmically. Therefore, given a set of equispaced nodes on $[a, b] \subseteq \mathbb{R}$, the fake CL nodes on $[-1, 1]$ are obtained by taking

$$S(x) = -\cos\left(\frac{x - a}{b - a}\pi\right), \tag{7}$$

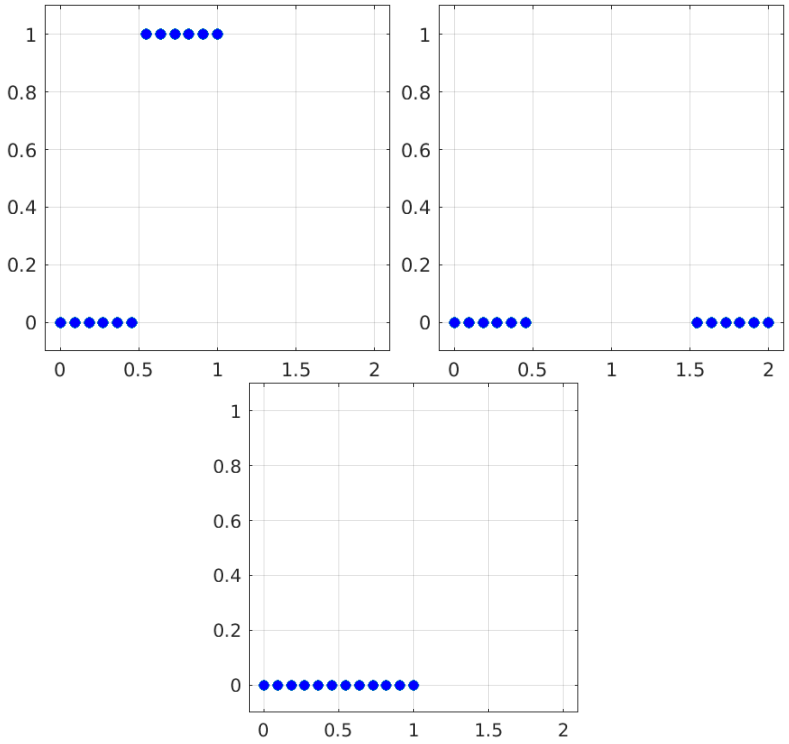

Figure 2: Top left: fake nodes mapped via S. Top right: nodes for the VSDKs setting in $\mathbb{R}^2$. Bottom: original set of equispaced nodes.

as mapping function.

When dealing with multidimensional equispaced grid data, we can extend this idea by considering tensor-product CL grids. Let

$$\Omega = \bigotimes_{j=1}^{d} [a_j, b_j] \subseteq \mathbb{R}^d,$$

be a rectangular domain where $a_j < b_j$. Let

$$X_{N_j} \coloneqq \left\{ a_j + \frac{i-1}{N_j - 1}(b_j - a_j),\ \ i = 1, \ldots, N_j \right\},$$

$j = 1, \ldots, d$, be d sets of equispaced points. Then, we associate to

$$X_N = \prod_{j=1}^{d} X_{N_j} \subset \Omega,$$

the usual set of function values F_N. Finally, once we obtain the nodes $S(X_{N_j})$, $j = 1, \ldots, d$, by using (7), we simply construct the tensor product interpolant on Chebyshev grids, see e.g. [3, 16].

5.2. The fake Padua approach

Here, we consider polynomial interpolation of total degree on $\Omega = [-1,1]^2$ (any two-dimensional finite rectangular domain could be considered). We recall that the basis for bivariate polynomials of total degree n has cardinality

$$N = \binom{n+2}{2} = \frac{(n+1)(n+2)}{2},$$

and that the interpolant $P_f(\boldsymbol{x}) \in \text{span}\{x_1^i x_2^j, i,j = 0,\dots,n,\ \ i+j \le n\}$.

As a consequence, in this framework we shall restrict to unisolvent sets of nodes of cardinality N. More precisely, we take the set

$$X_N = \left\{ \left(\frac{2(i-1)}{n} - 1, \frac{2(j-1)}{n+1} - 1 \right), \begin{array}{c} i = 1,\dots,n+1 \\ j = 1,\dots,n+2 \end{array}, \begin{array}{c} i+j \equiv 0 \\ (\text{mod } 2) \end{array} \right\}, \quad (8)$$

which is extracted from a $N = (n+1)\times(n+2)$ equispaced grid on Ω. There is a unique polynomial of degree n that interpolates the function f at the nodes X_N but, as in the univariate case, such interpolation is affected by errors that increase exponentially with n.

Considering the fake nodes approach, an optimal set of nodes is given by the so-called *Padua points*. There are four families of Padua points. Here we consider the first family, being the others obtained by counterclockwise rotations of considered one, which is defined as follows:

$$P_N = \left\{ \varphi\left(\frac{k\pi}{n(n+1)} \right), k = 0,\dots,n(n+1) \right\},$$

where

$$\varphi(t) = (-\cos((n+1)t), -\cos(nt)), \quad t \in [0,\pi].$$

is a closed parametric curve in Ω, and it is a special case of Lissajous curves; see [7, 18, 23].

To our aims, the fundamental property of the set P_N is that its Lebesgue constant is of minimal growth, since it has been proven to be $O(\log N)^2$. We propose to use the fake nodes approach with the map $S : \Omega \longrightarrow \Omega$ defined as

$$S(x,y) = \left(-\cos\left(\pi\frac{x+1}{2}\right), -\cos\left(\pi\frac{y+1}{2}\right) \right).$$

In fact, it is easy to prove that the fake nodes $S(X_N)$ are exactly the Padua points of the first kind P_N. Then, we can construct a new interpolant at the fake nodes $S(X_N)$ as described in (2).

Finally, we observe that if we use the condition $i+j \equiv 1 \pmod 2$ in (8), the second kind Padua points will result. To obtain the $\pi/2$-counterwise rotation of the Padua points it is sufficient to swap the two coordinates in (8).

We conclude this section by showing a pseudocode of the fake nodes approach, which works for every different choice of the injective map S and of the selected basis (see Algorithms 1–2).

Algorithm 1. Interpolation.

Inputs:
$X_N = \{\boldsymbol{x}_i,\ i = 1, \dots, N\} \subseteq \Omega \subset \mathbb{R}^d$ interpolation nodes;
$\mathcal{F}_N = \{f(\boldsymbol{x}_i),\ i = 1, \dots, N\}$ values at interpolation nodes;
$\mathcal{B} \quad = \{B_i,\ i = 1, \dots, N\}$ interpolation basis;
$\boldsymbol{x} \in \Omega$ generic evaluation node.
Main procedure:
1. Compute the matrix $\mathsf{A}_{ij} = B_i(\boldsymbol{x}_j)$, $i, j = 1, \dots, N$.
2. Compute the coefficients α_i, $i = 1, \dots, N$ by solving the linear system (1).
3. Evaluate the interpolant at $\boldsymbol{x} \in \Omega$, i.e. compute $P_f(\boldsymbol{x})$ by using (2).

Outputs: $P_f(\boldsymbol{x})$ for $\boldsymbol{x} \in \Omega$.

Algorithm 2. Fake nodes interpolation.

Inputs:
$X_N = \{\boldsymbol{x}_i,\ i = 1, \dots, N\} \subseteq \Omega \subset \mathbb{R}^d$ interpolation nodes;
$\mathcal{F}_N = \{f(\boldsymbol{x}_i),\ i = 1, \dots, N\}$ values at interpolation nodes;
$\mathcal{B} \quad = \{B_i,\ i = 1, \dots, N\}$ interpolation basis;
$S : \Omega \longrightarrow \mathbb{R}^d$ injective map;
$\boldsymbol{x} \in \Omega$ generic evaluation node.
Main procedure:
1. Compute $S(X_N) = \{S(\boldsymbol{x}_i),\ i = 1, \dots, N\}$.
2. Compute $S(\boldsymbol{x})$.
3. Compute $R_f(\boldsymbol{x})$ as the output of Algorithm 1 using as inputs: $S(X_N)$ as interpolation nodes, $\mathcal{F}_N$ as values, $\mathcal{B}$ as basis, $S(\boldsymbol{x})$ as evaluation node.

Outputs: $R_f(\boldsymbol{x})$ for $\boldsymbol{x} \in \Omega$.

We now focus on numerical experiments proposing various techniques and comparing them.

6. Numerical Experiments

In the following experiments we point out three important aspects of the fake nodes method. Precisely:

1. The versatility of the fake nodes approach with respect to different basis functions. In doing this, we focus on discontinuous test functions and therefore we use the S-Gibbs map.

2. The applicability of the fake nodes approach to medical imaging. We test this via polynomial least squares.

3. The ability to mitigate the Runge effect. Also in this case we drive our attention towards polynomial bases.

6.1. Versatility of the fake nodes approach

In this subsection, our main scope consists in numerically showing the flexibility of the fake nodes approach, meaning that it can be applied to *all* basis functions. Indeed, mapping data without the need of resampling can be used as a kind of *black box* for any interpolation or approximation procedure.

We test three approximation techniques, i.e. three different basis functions:

- Polynomials: since for a general set of scattered data we might not have unisolvent sets, we focus on polynomial least squares and we fix the degree equal to 4. For the implementation (see e.g. [23, 24, 33]) we refer the reader to the MATLAB packages available at the CAA research group homepage `https://www.math.unipd.it/~marcov/CAA.html` and to the the GitHub repositories by Wolfgang Erb and Marco Vianello available at `https://github.com/WolfgangErb` and `https://github.com/marcovianello`.

- Kernels: in what follows, for computing the kernel-based interpolant, we fix the Matérn kernel, given by

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{e}^{-\gamma\|\boldsymbol{x}-\boldsymbol{y}\|_2}.$$

  For its implementation, we provide a free PYTHON software available at `https://github.com/pog87/FakeNodes2D`. The shape parameter is set as $\gamma = 0.5$.

- Nearest-Neighbor (n) interpolation: for each evaluation point, the algorithm returns the function value associated to the nearest node, yielding a piecewise-constant interpolant. It is discontinuous for its nature and thus it is not affected by the Gibbs phenomenon. As a consequence, it is a challenging test for the fake nodes tool. For its implementation, we use the PYTHON function `griddata` of the package `scipy`.

To test the efficacy of the S-Gibbs map, we take the following test function:

$$f(x_1, x_2) = \begin{cases} \sin(x_1 + x_2^2), & \text{if} \quad x_1^2 + x_2^2 - 0.4^2 < 0, \\ 1, & \text{if} \quad x_1^2 + x_2^2 - 0.4^2 \geq 0, \end{cases}$$

and we sample it at $N = \{9, 81, 289, 1089, 4225\}$ grid data on $[-1, 1]^2$. The accuracy is tested by evaluating the Mean Square Error (MSE) on a grid of 40^2 evaluation points. The results are reported in Figure 3. We can note that the error with the fake nodes is sensibly reduced with respect to the standard approach.

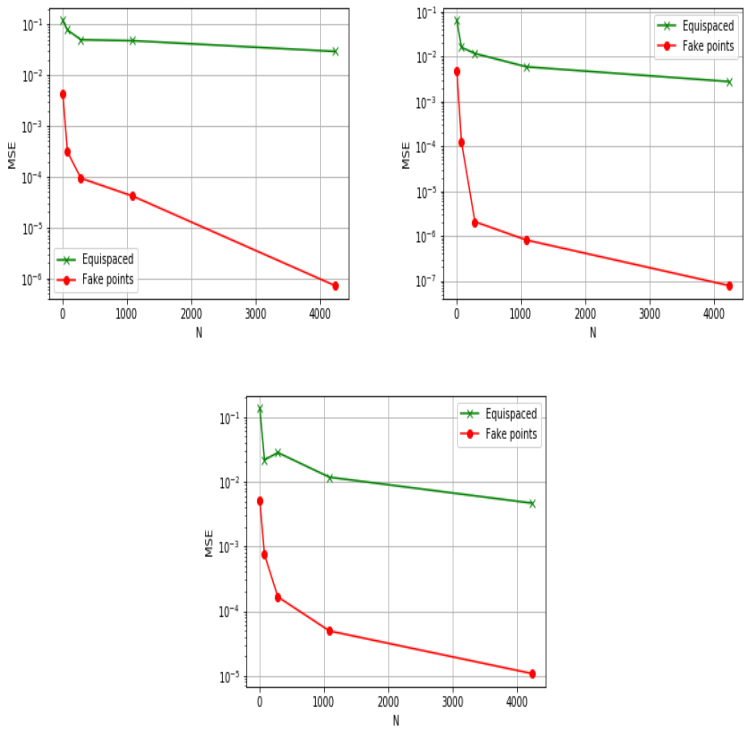

Figure 3: The MSE by varying N with polynomials (top left), kernels (top right) and NN (bottom). The green stars represent the standard bases, while the results for mapped bases are plotted with the dotted red line.

6.2. Applicability to medical imaging

As a second example for testing the S-Gibbs algorithm, we take the Shepp-Logan phantom. This example also stresses the importance of the present approach for medical imaging. The Shepp-Logan phantom is plotted in Figure 4 (left) and its size is 256×256. We then subsample it on $N = \{32^2, 48^2, 64^2, 96^2, 128^2\}$ pixels (see e.g. Figure 4 right) and we evaluate the performances of the methods in reconstructing the original phantom. A graphical example is plotted in Figure 5, where we use the least squares polynomials with and without the use of fake nodes. The MSE is depicted in Figure 6. Once more, we can note the robustness of the presented approach in reducing the Gibbs effect.

6.3. Mitigating the 2D Runge effect

Let us consider the bivariate Runge function $f : [-1, 1]^2 \longrightarrow \mathbb{R}$, defined as

$$f(x, y) = \frac{1}{1 + 5(x^2 + y^2)}.$$

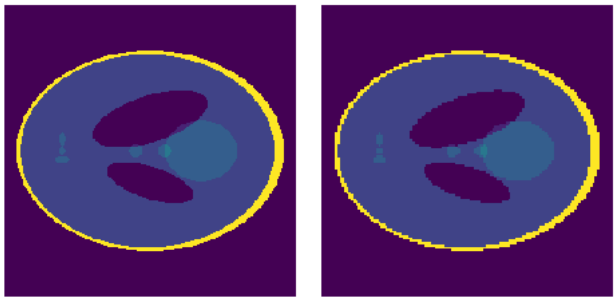

Figure 4: Original Shepp-Logan (left), subsampled Shepp-Logan (128×128) pixels (right).

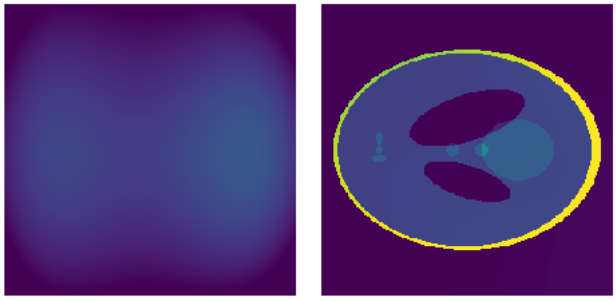

Figure 5: Left to right: standard polynomial reconstruction, fake polynomial reconstruction on 128×128 pixels.

In order to test the lines approach presented in Section 5.1, we compare it with the standard tensor-product polynomial reconstruction at equispaced grids and with the reconstruction obtained by resampling on Chebyshev-Lobatto grids. In Figure 7, we display the results considering a 13×15 starting grid, while in Figure 8 we observe the asymptotic behavior of the considered methods by means of an increasing sequence of $n \times n$ grids. Considering the fake Padua approach of Section 5.2, we test it with the multivariate polynomial reconstruction at equispaced $n \times (n+1)$ grids and with the reconstruction at Padua points with resampling. In Figure 9, we display the results considering a 10×10 starting grid, while in Figure 10 we observe the asymptotic behavior of the considered methods by means of an increasing sequence of $n \times (n+1)$ grids.

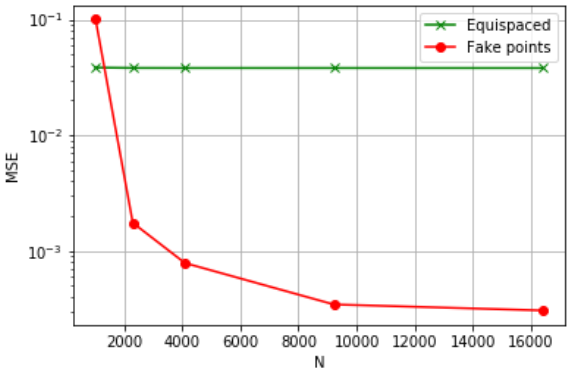

Figure 6: The MSE by varying N with polynomials for the Shepp-Logan phantom. The green stars represent the standard bases, while the results for mapped bases are plotted with the dotted red line.

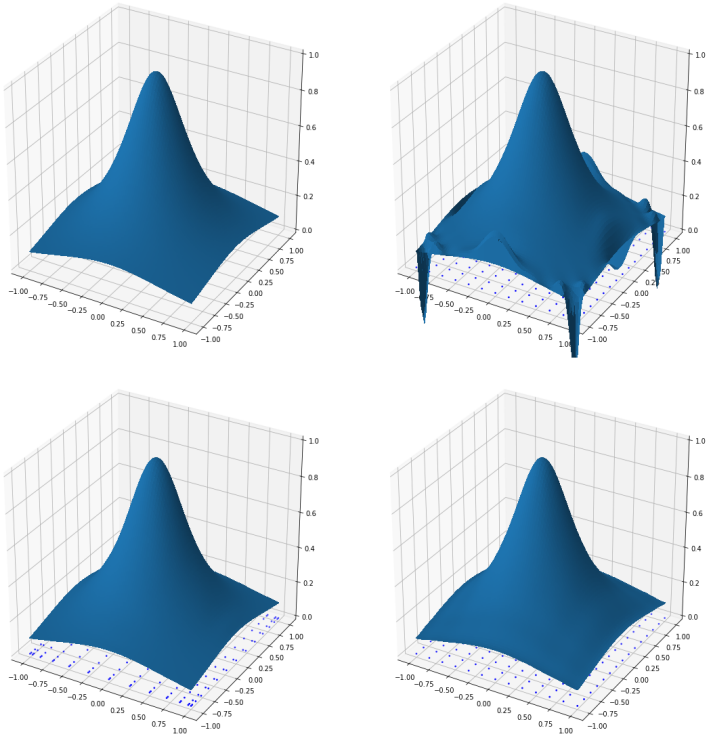

Figure 7: Results of different interpolation schemes considering a 13×15 grid: the original function (top left), the tensor-product reconstruction at the equispaced grid (top right), the reconstruction with resampling at the CL grid (bottom left) and the fake lines approach at the equispaced grid without resampling (bottom right).

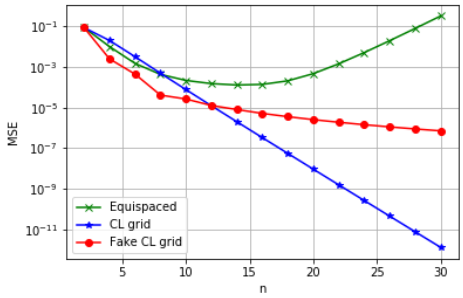

Figure 8: The MSE by varying n in the interpolation grid $n \times n$. The green crosses represent the tensor-product reconstruction at the equispaced grid, the blue stars represent the reconstruction with resampling at the CL grid, while the results for mapped bases are plotted with the dotted red line.

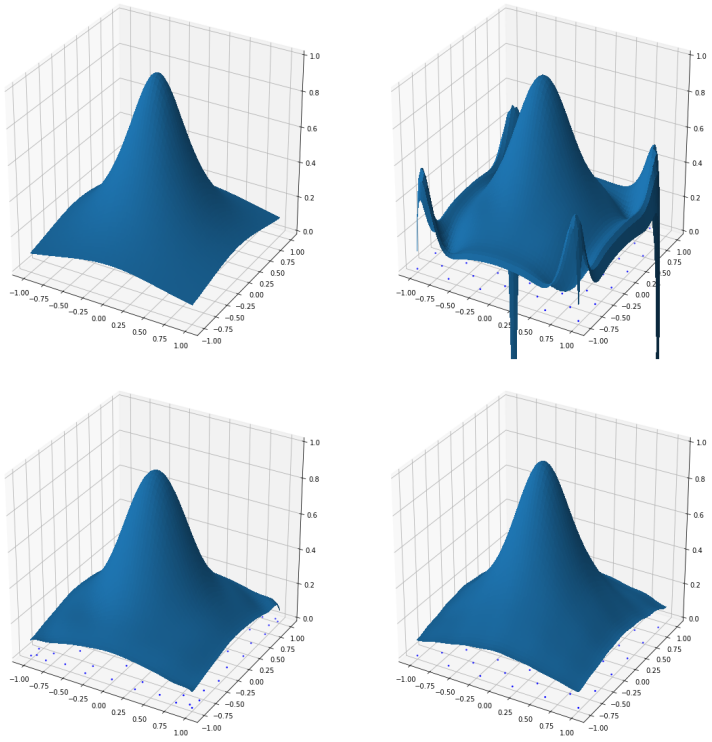

Figure 9: Results of different interpolation schemes for $n = 10$: the original function (top left), the multivariate polynomial reconstruction at the equispaced grid (top right), the reconstruction with resampling at the Padua points (bottom left) and the fake Padua approach at the equispaced grid without resampling (bottom right).

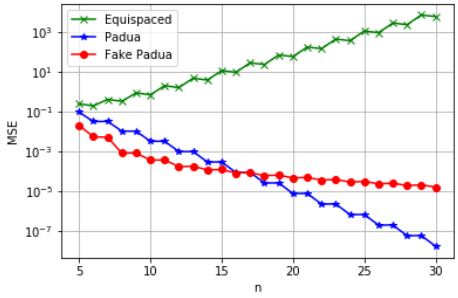

Figure 10: The MSE by varying n in the interpolation grid $n \times n$. The green crosses represent the multivariate polynomial reconstruction at the equispaced grids, the blue stars represent the reconstruction with resampling at the Padua points, while the results for mapped bases are plotted with the dotted red line.

7. Conclusions

We presented a numerical scheme that can be used for multivariate scattered data interpolation methods and that allows to overcome many drawbacks, such as to prevent non physical oscillations in the reconstruction process, when functions with steep gradients or discontinuities are involved. It takes advantage of working with different bases and of being easy to implement. Work in progress consists in studying its efficacy in the context of collocation schemes for solving PDEs and in investigating its applicability in the framework of machine learning.

Acknowledgments

This research has been accomplished within Rete ITaliana di Approssimazione (RITA), partially funded by GNCS-INδAM and through the European Union's Horizon 2020 research and innovation programme ERA-PLANET, grant agreement no. 689443, via the GEOEssential project.

References

[1] B. Adcock, R.B. Platte, *A mapped polynomial method for high-accuracy approximations on arbitrary grids*, SIAM J. Numer. Anal. **54** (2016), 2256–2281.

[2] S. Amat, J. Ruiz, J.C. Trillo, D.F. Yáñez, *Analysis of the Gibbs phenomenon in stationary subdivision schemes*, Appl. Math. Lett. **76** (2018), 157–163.

[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] A. Bayliss, E. Turkel, *Mappings and accuracy for Chebyshev pseudo-spectral approximations*, J. Comput. Phys. **101** (1992), 349–359.

[5] J.P. Berrut, S. De Marchi, G. Elefante, F. Marchetti, *Treating the Gibbs phenomenon in barycentric rational interpolation via the S-Gibbs algorithm*, to appear on Appl. Math. Letters. 2019.

[6] L. Bos, D. De Marchi, K. Hormann, *On the Lebesgue constant of Berrut's rational interpolant at equidistant nodes*, J. Comput. Appl. Math. **236** (2011), 504–510.

[7] L. Bos, S. De Marchi, M. Vianello, *Polynomial approximation on Lissajous curves in the d−cube*. Appl. Numer. Math. **116** (2017), pp. 47–56.

[8] M. Bozzini, L. Lenarduzzi, M. Rossini, R. Schaback, *Interpolation with variably scaled kernels*, IMA J. Numer. Anal. **35** (2015), 199–219.

[9] S.C. Brenner, L.R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer New York, 1994.

[10] C. Brezinski, *Biorthogonality and its Applications to Numerical Analysis*, Marcel Dekker Inc., New York, 1992.

[11] L. Brutman, *On the Lebesgue function for polynomial interpolation*, SIAM J. Numer. Anal. **15** (1978), 694–704.

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

[13] E.W. Cheney, *Introduction to Approximation Theory*, AMS Chelsea Pub, New York, 2000.

[14] P.C. Curtis Jr., *N-parameter families and best approximation*, Pacific J. Math. **9** (1959), pp. 1013–1027.

[15] P.J. Davis, *Interpolation and Approximation*, Dover Publications, New York, 1975.

[16] C. De Boor, *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.

[17] S. De Marchi, *Polynomials arising in factoring generalized Vandermonde determinants: an algorithm for computing their coefficients*, Math. Comput. Modelling **34** (2001), 271–281.

[18] 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.

[19] S. De Marchi, W. Erb, F. Marchetti, E. Perracchione, M. Rossini, *Shape-Driven Interpolation with Discontinuous Kernels: Error Analysis, Edge Extraction and Applications in MPI*, submitted, 2019.

[20] S. De Marchi, F. Marchetti, E. Perracchione, *Jumping with Variably Scaled Discontinuous Kernels (VSDKs)*, submitted, 2019.

[21] S. De Marchi, F. Marchetti, E. Perracchione, D. Poggiali, *Polynomial interpolation via mapped bases without resampling*, J. Comput. Appl. Math., **364** (2020), 112347–12.

[22] P. Dencker, W. Erb, *A unifying theory for multivariate polynomial interpolation on general Lissajous-Chebyshev nodes*, preprint 2019.

[23] W. Erb, *Bivariate Lagrange interpolation at the node points of Lissajous curves - the degenerate case*, Appl. Math. Comput. **289** (2016), 409–425.

[24] W. Erb, C. Kaethner, M. Ahlborg, T.M. Buzug, *Bivariate Lagrange interpolation at the node points of non-degenerate Lissajous curves*, Numer. Math. **133** (2016), 685–705.

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

[26] M.S. Floater, *Polynomial interpolation on interlacing rectangular grids*, J. Approx. Theory **222** (2017), pp. 64–73.

[27] D. Gottlieb, C.W. Shu, *On the Gibbs phenomenon and its resolution*, SIAM Review **39** (1997), pp. 644–668.

[28] A. Haar, *Die Minkowskische Geometrie und die Annäherung an stetige Funktionen*, Math. Ann. **18** (1918), pp. 294–311.

[29] D. Kalman, *The Generalized Vandermonde Matrix*. Mathematics Magazine, 57(1), 15-21 (1984). doi:10.2307/2690290

[30] D. Kosloff, H. Tal-Ezer, *A modified Chebyshev pseudospectral method with an $O(N^{-1})$ time step restriction*, J. Comput. Phys. **104** (1993), 457–469.

[31] J.C. Mairhuber, *On Haar's theorem concerning Chebychev approximation problems having unique solutions*, Proc. Amer. Math. Soc. **7** (1956), 609–615.

[32] G. Mastroianni, D. Occorsio, *Optimal systems of nodes for Lagrange interpolation on bounded intervals. A survey*, J. Comput. Appl. Math. **134** (2001) pp. 325–341.

[33] F. Piazzon, A. Sommariva, M. Vianello, *Caratheodory-Tchakaloff Least Squares*, Sampling Theory and Applications 2017, IEEE Xplore Digital Library, 672–676.

[34] M. Rossini, *Interpolating functions with gradient discontinuities via variably scaled kernels*, Dolom. Res. Notes Approx. **11** (2018), 3–14.

[35] L. Romani, M. Rossini, D. Schenone, *Edge detection methods based on RBF interpolation*, J. Comput. Appl. Math. **349** (2019), 532–547.

[36] C. Runge, *Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten*, Zeit. Math. Phys. **46** (1901), 224–243.

[37] L.L. Schumaker, *Spline Functions: Basic Theory*, John Wiley & Sons, New York, 1981.

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