Polynomial interpolation via mapped bases without resampling

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

Abstract

In this work we propose a new method for univariate polynomial interpolation based on what we call *mapped bases*. As theoretically shown, constructing the interpolating function via the mapped bases, i.e. in the *mapped space*, turns out to be equivalent to map the nodes and then construct the approximant in the *classical* form without the need of resampling. In view of this, we also refer to such mapped points as "fake" nodes. Numerical evidence confirms that such scheme can be applied to mitigate *Runge* and *Gibbs* phenomena.

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

1. Introduction

Univariate polynomial interpolation has been widely studied in the past and nowadays finds many applications, e.g. to scientific computing and time series analysis. Recent research focuses on constructing admissible sets of points in higher dimensions so that the interpolation problem is well-posed, see e.g. [26]. Further studies are also devoted to build sets of data that *minimize* the *Lebesgue constant* for both multivariate interpolation problems and rational univariate approximants, see e.g. [7, 8]. In this direction, many works based on conformal maps have already been developed; for a general overview, we refer the reader to [1, 6] and references within.

However, this might not be effective enough for applications, indeed choosing *ad hoc* sets of nodes leads to resampling the (unknown) function. Therefore, this is not always possible and one only disposes of a given set of nodes and related function values sampled at those points. Alternatively to resampling,

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*)

one could consider the idea of extracting "good" nodes from the given samples. In this view, an example are the so-called *mock-Chebyshev* points, which present a quasi-Chebyshev distribution in the interval and they are extracted from equispaced samples [9].

We here adopt a different solution and we propose the so-called mapped bases approach, which is equivalent to build (under certain restrictions) an *arbitrary* set of interpolation nodes with the sought properties, e.g. Chebyshev nodes for reducing the *Runge* phenomenon [11, 12, 30, 31]. The main advantage is that the resampling is not needed, i.e. we reduce to solving a classical interpolation problem on these *mapped* or *fake nodes* and then we apply the same map to the evaluation points.

As another immediate application, we focus on reducing the Gibbs oscillations; refer e.g. to [2, 3, 18, 21, 22, 24] for a general overview. Precisely, inspired by recent studies in the context of meshfree methods [10, 19, 28], by selecting a discontinuous map, we can naturally reconstruct jumps with the monomial/lagrangian basis.

The paper is organized as follows. In Section 2, after briefly reviewing the basics of univariate polynomial interpolation, we describe our interpolation method based on fake nodes. In Section 3, we present algorithms devoted to the construction of robust mapping functions for our interpolation strategy. Numerical experiments and comparisons with standard approaches used to mitigate Runge and Gibbs phenomena, are presented in Section 4. The last section deals with conclusions and future work.

Finally, we point out that the PYTHON software for interpolating at the fake nodes is available for the scientific community in [20].

2. From polynomial interpolation to mapped bases

We first review the basics of univariate interpolation; refer e.g. to [15].

Let $\Omega = [a, b] \subset \mathbb{R}$ be an interval and let $\mathcal{X}_{n+1} = \{x_i\}_{i=0,\dots,n} \subset \Omega$ be a set of distinct nodes (also called data sites). The aim is to reconstruct a function $f : \Omega \longrightarrow \mathbb{R}$ given its samples at the nodes, i.e. given $\mathcal{F}_{n+1} = \{f_i = f(x_i)\}_{i=0,\dots,n}$. More precisely, in the interpolation setting, let Π_n be the set of polynomials of degree n, we seek for the polynomial $P_{n,f}$ such that:

$$P_{n,f}(x_i) = f_i, \quad i = 0, \dots, n. \tag{1}$$

$P_{n,f}$ can be expressed in the monomial basis $\mathcal{M} = \text{span}\{1, x, \dots, x^n\}$ as

$$P_{n,f}(x) = \sum_{i=0}^{n} c_i x^i.$$

The interpolation problem (1) then leads to finding the vector of coefficients $\boldsymbol{c} = (c_0, \dots, c_n)^\intercal$. As long as the nodes are distinct, the latter is uniquely determined by solving the linear system

$$V\boldsymbol{c} = \boldsymbol{f}, \tag{2}$$

where $V = V(x_0, \ldots, x_n) \in \mathbb{R}^{n+1} \times \mathbb{R}^{n+1}$ is the well-known *Vandermonde matrix* and $\boldsymbol{f} = (f_0, \ldots, f_n)^\intercal$.

The polynomial interpolant can be written in the so-called *Lagrange basis* $\mathcal{L} = \operatorname{span}\{\ell_0, \ldots, \ell_n\}$ so that

$$P_{n,f}(x) = \sum_{i=0}^{n} f_i \ell_i(x),$$

for $x \in \Omega$ and where

$$\ell_i(x) = \frac{\det(V_i(x))}{\det(V)} = \prod_{\substack{0 \le j \le n \\ j \neq i}} \frac{x - x_j}{x_i - x_j}, \tag{3}$$

is the i-th Lagrange polynomial and $V_i(x) \coloneqq V(x_0, \ldots, x_{i-1}, x, x_{i+1}, \ldots, x_n)$.

Most of recent research on the topic orbits around the *Lebesgue* constant defined by:

$$\Lambda_n(\Omega) = \max_{x \in \Omega} \sum_{i=0}^{n} |\ell_i(x)|, \tag{4}$$

and related upper bounds. The relevance of investigating in that field follows from the fact that it is a measure of stability and accuracy. Indeed, letting

$$E_n(f) \coloneqq \max_{x \in \Omega} |f(x) - P_{n,f}(x)| = \|f - P_{n,f}\|_\Omega,$$

for $f \in C(\Omega)$ we have

$$E_n(f) \le (1 + \Lambda_n(\Omega)) E_n^\star(f), \tag{5}$$

where $E_n^\star(f)$ is the best polynomial approximation error in the space Π_n [11].

Moreover, let $\mathcal{L}_n : C(\Omega) \longrightarrow \Pi_n$ be the linear and bounded operator that associates $f \in C(\Omega)$ to its interpolant $P_{n,f} \in \Pi_n$ on $\mathcal{X}_{n+1}$, then

$$\Lambda_n(\Omega) = \sup_{\substack{f \in C(\Omega) \\ f \neq 0}} \frac{\|\mathcal{L}_n(f)\|_\Omega}{\|f\|_\Omega}, \tag{6}$$

is the operator norm of $\mathcal{L}_n$ with respect to $\|\cdot\|_\Omega$.

Based on this, we now introduce a method that changes the given interpolation problem without resampling the unknown function f at other interpolation nodes. The idea of mapped polynomials is not new. Indeed, such kinds of methods have been used in the context of spectral schemes for PDEs. Here the map is used to mitigate the oscillations due to Gibbs and Runge phenomena. For examples of well-known maps refer to e.g. [23, 25]. However, for our purposes, that are devoted especially to applications when resampling cannot be performed, we consider a generic map $S : \Omega \longrightarrow \mathbb{R}$. We investigate the restrictions for the choice of S in Subsection 2.1.

For $\tilde{x} \in S(\Omega)$, we can compute the polynomial $P_{n,g} : S(\Omega) \longrightarrow \mathbb{R}$ interpolating the function values $\mathcal{F}_{n+1}$ at the *fake nodes* $S(\mathcal{X}_{n+1}) = \{S(x_i) = \tilde{x}_i\}_{i=0,\dots,n} \subset S(\Omega)$ defined by

$$P_{n,g}(\tilde{x}) = \sum_{i=0}^{n} c_i \tilde{x}^i,$$

for some function $g : S(\Omega) \longrightarrow \mathbb{R}$ so that it belongs to $C^k(\Omega)$, $k \geq 0$ and

$$g_{|S(\mathcal{X}_{n+1})} = f_{|\mathcal{X}_{n+1}}.$$

Hence, for $x \in \Omega$ we are interested in studying the function

$$R^s_{n,f}(x) \coloneqq P_{n,g}(S(x)) = \sum_{i=0}^{n} c_i S(x)^i. \tag{7}$$

The function $R^s_{n,f}$ in (7) can be considered as an interpolating function at the original set of nodes $\mathcal{X}_{n+1}$ and data values $\mathcal{F}_{n+1}$, which is a linear combination of the basis functions $\mathcal{S}_n \coloneqq \{(S(x))^i\}_{i=0,\dots,n}$. As far as we know, a similar approach has been mentioned in [5], without being later worked out. We thus now study the proposed approach from different points of view.

In summary, the analysis of the interpolation process can be performed in the following equivalent ways.

- The *mapped bases* approach on Ω: we interpolate f on the set $\mathcal{X}_{n+1}$ via $R^s_{n,f}$ in the function space $\mathcal{S}_n$.
- The *fake nodes* approach on $S(\Omega)$: we interpolate g on the set $S(\mathcal{X}_{n+1})$ via $P_{n,g}$ in the polynomial space Π_n.

We discuss the two approaches below.

2.1. The mapped bases approach

From now on, we use the abridged notation $S(x_i) = S_i$. The first topic to investigate is whether the choice of the map S that determines the finite-dimensional space $\mathcal{S}_n$ is arbitrary or not. We say that S is *admissible* if the resulting interpolation process admits a unique solution, which is verified as long as the Vandermonde-like matrix

$$V^s = V^s(x_0, \dots, x_n) = V(S_0, \dots, S_n),$$

is invertible, that is

$$\det(V^s) = \prod_{0 \leq i < j \leq n} (S_j - S_i) \neq 0.$$

In what follows, we restrict to admissible maps and characterizations of admissible functions are given in the following proposition.

Proposition 2.1 *The function S is admissible if and only if for any $0 \leq i, j \leq n$, $i \neq j$ we have*

$$S_i \neq S_j.$$

In other words, S is injective in $\mathcal{X}_{n+1}$.

Proof: We know that $\det(V^s)$ is non-zero if and only if $S_j - S_i \neq 0$. ∎

Remark 2.1 *We point out that we can easily write*

$$\det(V^s) = \sigma(S, \mathcal{X})\det(V),$$

where V is the classical Vandermonde matrix (see (2)) and

$$\sigma(S, \mathcal{X}) \coloneqq \prod_{0 \leq i < j \leq n} \frac{S_j - S_i}{x_j - x_i}.$$

This presents some similarities with the framework of generalized Vandermonde determinants and Schur functions (cf. [17]).

Let $(\Upsilon, \|\cdot\|_\Omega)$ be a normed function space, which contains only real-valued functions on $\Omega = [a, b]$ and let us assume that the function f belongs to Υ which is so that

$$\mathcal{S}_0 \subset \mathcal{S}_1 \subset \cdots \subset \mathcal{S}_n \subset \cdots \subset \Upsilon.$$

In that case we have the following proposition, see e.g. [27, Theorem I.1, p. 1].

Proposition 2.2 *If Υ is a normed space and $\mathcal{S}_n$ is a finite-dimensional subspace of Υ, given $f \in \Upsilon$, there exists $r_n^\star \in \mathcal{S}_n$ such that*

$$\|f - r_n^\star\|_\Omega = \min_{r_n \in \mathcal{S}_n} \|f - r_n\|_\Omega \coloneqq E_n^{s,\star}(f).$$

Moreover, we make the assumption that we can choose the function S in such a way that the space

$$\mathcal{S} \coloneqq \bigcup_{n \in \mathbb{N}} \mathcal{S}_n,$$

is dense in Υ, i.e. for any $\epsilon > 0$ there exists $v_\epsilon \in \mathcal{S}_n$ such that $\|f - v_\epsilon\|_\Omega < \epsilon$. With the considered assumption, we have that $E_n^{s,\star}(f) \longrightarrow 0$ as $n \longrightarrow \infty$ (see e.g. [27, Theorem 1.1, p. 11]).

Thus, we can associate to the interpolant $R_{n,f}^s \in \mathcal{S}_n$ its Lagrange form

$$R_{n,f}^s(x) = \sum_{i=0}^{n} f_i \ell_i^s(x),$$

where, in analogy with (3),

$$\ell_i^s(x) \coloneqq \frac{\det(V_i^s(x))}{\det(V^s)} = \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{S(x) - S_j}{S_i - S_j}, \tag{8}$$

where $V_i^s(x) = V_i(S(x))$.

Consequently, we can consider the Lebesgue constant $\Lambda_n^s(\Omega)$ associated to the Lagrange basis $\mathcal{L}^s = \{\ell_0^s, \ldots, \ell_n^s\}$ and to the interpolation operator $\mathcal{L}_n^s : \Upsilon \longrightarrow \mathcal{S}_n$, as previously discussed. Then, $\Lambda_n^s(\Omega)$ is the operator norm of $\mathcal{L}_n^s$ with respect to $\|\cdot\|_\Omega$, that is

$$\Lambda_n^s(\Omega) = \sup_{\substack{v \in \Upsilon \\ v \neq 0}} \frac{\|\mathcal{L}_n^s(v)\|_\Omega}{\|v\|_\Omega},$$

and hence

$$\|f - R_{n,f}^s\|_\Omega \leq (1 + \Lambda_n^s(\Omega)) E_n^{s,\star}(f),$$

In order to present a bound for $\Lambda_n^s(\Omega)$ in terms of the Lebesgue constant related to the classical polynomial interpolation, we need the following result.

Proposition 2.3 *Let ℓ_i be the classical i-th Lagrange polynomial and let ℓ_i^s be defined as in (8). For all $x \in \Omega$, $x \neq x_j$ with $j \neq i$, we have*

$$\ell_i^s(x) = \gamma_i(x)\ell_i(x),$$

where

$$\gamma_i(x) \coloneqq \frac{\det(V_i^s(x))}{\sigma(S, \mathcal{X})\det(V_i(x))},$$

with $\sigma(S, \mathcal{X})$ as in Remark 2.1. Moreover,

$$\gamma_i(x) = \frac{\beta_i(x)}{\alpha_i},$$

with

$$\alpha_i \coloneqq \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{S_i - S_j}{x_i - x_j}, \quad \beta_i(x) \coloneqq \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{S(x) - S_j}{x - x_j}.$$

Proof: For $x = x_j$, with $j \neq i$, $\ell_i^s(x_j) = \ell_i(x_j) = 0$. Then let $x \neq x_j$; from (3) and (8) we have that

$$\frac{\ell_i^s(x)}{\ell_i(x)} = \frac{\det(V_i^s(x))\det(V)}{\det(V_i(x))\det(V^s)} = \frac{\det(V_i^s(x))}{\sigma(S, \mathcal{X})\det(V_i(x))} \coloneqq \gamma_i(x).$$

If $i \neq j$, then $\ell_i^s(x_j) = 0$. Let $x \neq x_j$, we can write

$$\begin{aligned}
\ell_i^s(x) &= \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{S(x) - S_j}{S_i - S_j}, \\
&= \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{x - x_j}{x_i - x_j} \cdot \frac{x_i - x_j}{x - x_j} \cdot \frac{S(x) - S_j}{S_i - S_j}, \\
&= \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{x - x_j}{x_i - x_j} \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{S(x) - S_j}{x - x_j} \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{x_i - x_j}{S_i - S_j}.
\end{aligned}$$

By defining

$$\alpha_i \coloneqq \prod_{\substack{0\le j\le n\\ j\neq i}} \frac{S_i - S_j}{x_i - x_j}, \quad \beta_i(x) \coloneqq \prod_{\substack{0\le j\le n\\ j\neq i}} \frac{S(x) - S_j}{x - x_j},$$

we obtain

$$\ell_i^s(x) = \frac{\beta_i(x)}{\alpha_i}\ell_i(x).$$

∎

Theorem 2.4 *We have that*

$$\Lambda_n^s(\Omega) \le \left(\frac{L}{D}\right)^n \Lambda_n(\Omega),$$

where

$$L = \max_j \max_{x\in\Omega} \left|\frac{S(x) - S_j}{x - x_j}\right|,$$

$$D = \min_i \min_{j\neq i} \left|\frac{S_i - S_j}{x_i - x_j}\right|,$$

and $\Lambda_n(\Omega)$ *is the Lebesgue constant in* (4).

Proof: Taking into account Proposition 2.3, we consider the absolute values

$$|\ell_i^s(x)| = \frac{|\beta_i(x)|}{|\alpha_i|}|\ell_i(x)|.$$

We proceed by giving an upper bound for $|\beta_i|$:

$$|\beta_i(x)| = \prod_{\substack{0\le j\le n\\ j\neq i}} \left|\frac{S(x) - S_j}{x - x_j}\right| \le \prod_{\substack{0\le j\le n\\ j\neq i}} L_i^j,$$

where

$$L_i^j \coloneqq \max_{x\in\Omega} \left|\frac{S(x) - S_j}{x - x_j}\right|.$$

Thus,

$$|\beta_i(x)| \le L_i^n,$$

with $L_i \coloneqq \max_{j\neq i} L_i^j$. We then give a lower bound for $|\alpha_i|$:

$$|\alpha_i| = \prod_{\substack{0\le j\le n\\ j\neq i}} \left|\frac{S_i - S_j}{x_i - x_j}\right| \ge \prod_{\substack{0\le j\le n\\ j\neq i}} D_i = D_i^n,$$

where

$$D_i \coloneqq \min_{j\neq i} \left|\frac{S_i - S_j}{x_i - x_j}\right|.$$

We have that

$$|\ell_i^s(x)| \leq \left(\frac{L_i}{D_i}\right)^n |\ell_i(x)|.$$

Therefore, defining $L \coloneqq \max_i L_i$, $D \coloneqq \min_i D_i$ and considering the sum of the Lagrange polynomials, we obtain

$$\Lambda_n^s(\Omega) \leq \left(\frac{L}{D}\right)^n \Lambda_n(\Omega).$$

We conclude by observing that $\max_i \max_{j\neq i} L_i^j = \max_j L_j$. ∎

In the next subsection, we will show that the presented bound is largely pessimistic for certain choices of the map.

2.2. The fake nodes approach

The construction of the interpolating function $R_{n,f}^s$ is equivalent to build a polynomial interpolant at the fake nodes, as defined in (7). Therefore, in what follows we concisely analyze the parallelism with the polynomial interpolation problem in $S(\Omega)$.

If ℓ_i is the i-th Lagrange polynomial related to the set $S(\mathcal{X}_{n+1})$, then for $\tilde{x} \in S(\Omega)$, we have

$$\ell_i(\tilde{x}) = \prod_{\substack{0\leq j\leq n \\ j\neq i}} \frac{\tilde{x} - S(x_j)}{S(x_i) - S(x_j)},$$

and the Lebesgue constant is given by

$$\Lambda_n(S(\Omega)) = \max_{\tilde{x}\in S(\Omega)} \sum_{i=0}^{n} |\ell_i(\tilde{x})|. \tag{9}$$

For $x \in \Omega$, we observe that

$$\ell_i(\tilde{x}) = \ell_i(S(x)) = \prod_{\substack{0\leq j\leq n \\ j\neq i}} \frac{S(x) - S(x_j)}{S(x_i) - S(x_j)} = \ell_i^s(x).$$

As a consequence, we obtain

$$\Lambda_n^s(\Omega) = \Lambda_n(S(\Omega)),$$

and

$$\|f - R_{n,f}^s\|_\Omega = \|g - P_{n,g}\|_{S(\Omega)},$$

which implies in particular that

$$\|f - R_{n,f}^s\|_\Omega \leq (1 + \Lambda_n(S(\Omega)))E_n^\star(g).$$

Since we can suppose without loss of generality that g is a regular function, for an appropriate choice of the map S, and hence of the nodes $S(\mathcal{X}_{n+1})$, we might improve the results with respect to classical polynomial approximation in Ω. The main difficulties are in finding a *good* map. In the next section we thus propose two receipts for computing suitable maps that, as numerically shown later, enable us to naturally mitigate Runge and Gibbs phenomena.

3. Algorithms and good choices for the mapping function

In this section, we describe how, given an ordered set of distinct interpolation nodes $\mathcal{X}_{n+1} = \{x_i \in \Omega \mid x_0 = a,\ x_n = b,\ x_i < x_{i+1}\}$, we can effectively construct suitable maps S in $\Omega = [a, b]$. We remark that the function S has to be admissible according to Proposition 2.1. In what follows, we propose two different ways of constructing the map S and in doing so we deal with the Runge and Gibbs phenomena respectively.

Case 1: the Runge phenomenon

In order to prevent the appearance of the Runge phenomenon, our aim is to find a map S such that the resulting set of fake nodes $S(\mathcal{X}_{n+1})$ guarantees a stable interpolation process. The natural way is to consider the set of ordered *Chebyshev-Lobatto* (CL) nodes on Ω

$$\mathcal{C}_{n+1} = \{c_i\}_{i=0,\dots,n} = \left\{ \frac{a-b}{2} \cos\left(\pi \frac{i}{n}\right) + \frac{a+b}{2} \right\}_{i=0,\dots,n}.$$

Indeed, as well-known the Lebesgue constant of the CL nodes grows logarithmically with respect to n [11]. Finally, the construction of the map S on Ω is described in the following algorithm.

Algorithm 1. S-Runge.

Inputs: $\mathcal{X}_{n+1}$.

Main procedure:

1. Define the set of CL nodes $\mathcal{C}_{n+1}$.
2. For $x \in [x_i, x_{i+1}]$, $i = 0, \dots, n-1$, define S as the piecewise linear interpolant

$$S(x) = \beta_{1,i}(x - x_i) + \beta_{2,i},$$

where

$$\beta_{1,i} = \frac{c_{i+1} - c_i}{x_{i+1} - x_i}, \quad \beta_{2,i} = c_i.$$

Outputs: S.

Since the CL nodes are distinct, the obtained map is admissible. The presented procedure is robust and does not require any additional hypothesis on $\mathcal{X}_{n+1}$, indeed it works also for scattered data. However, other choices for particular sets of interpolation nodes could be considered. For instance, if $\mathcal{X}_{n+1} = \mathcal{E}_{n+1}$, where $\mathcal{E}_{n+1}$ is the ordered set of $n+1$ equispaced nodes on Ω, we can map the interpolation nodes analytically on the set $\mathcal{C}_{n+1}$ using

$$S(x) = \frac{a-b}{2} \cos\left(\pi \frac{x-a}{b-a}\right) + \frac{a+b}{2}. \tag{10}$$

In the next section, we show examples for both choices, i.e. scattered and equispaced sets of nodes.

Case 2: the Gibbs phenomenon

Let us suppose that the underlying function f presents jump discontinuities, whose positions and magnitudes are encoded in the set

$$\mathcal{D}_{m+1} \coloneqq \{(\xi_i, d_i) | \xi_i \in (a,b),\ \xi_i < \xi_{i+1},\ i = 0, \dots, m, \text{ and } d_i \coloneqq |f(\xi_i^+) - f(\xi_i^-)|\}.$$

In what follows, we suppose to know the discontinuities and the related jumps. Such assumption is not restrictive, indeed for the one dimensional case, many algorithms for detecting the discontinuity points are available. Among the huge existing literature on that, we refer the reader to e.g. [4, 29].

We remark that when discontinuities occur, in order to satisfy the interpolation conditions, the interpolant is forced to strongly vary near the jumps, hence the Gibbs phenomenon becomes evident. To overcome this problem, our strategy consists in constructing the map S in such a way that it sufficiently increases the gap between the node right before and the one right after the discontinuity. To accomplish this, we introduce what we will call the *shifting parameter* $k > 0$. As experimentally observed, its selection is not critical. Indeed, the resulting interpolation process is not sensitive to its chosen value, provided that it is sufficiently large, i.e. in such a way that in the mapped space the so-constructed function g has no steep gradients.

In the next algorithm, we formalize our idea for computing the map S in presence of discontinuities.

Algorithm 2. S-Gibbs

Inputs: $\mathcal{X}_{n+1}$, $\mathcal{D}_{m+1}$ and k.
Main procedure:

1. $\alpha_i \coloneqq k d_i,\ i = 0, \dots, m$.
2. Letting $A_i = \sum_{j=0}^{i} \alpha_j$, define S as follows:

$$S(x) = \begin{cases} x, & \text{for } x \in [a, \xi_0[, \\ x + A_i, & \text{for } x \in [\xi_i, \xi_{i+1}[,\ 0 \le i < m, \text{ or } x \in [\xi_m, b]. \end{cases}$$

Outputs: S.

Since the resulting fake nodes $S(\mathcal{X}_{n+1})$ are distinct, the so-constructed map is admissible.

We now provide examples devoted to show the robustness of our procedures.

4. Applications to Runge and Gibbs phenomena

In what follows, we show via the algorithms described in Section 3 that we are able to reduce the oscillating effects known as Runge and Gibbs phenomena. Our approach is compared to resampling with Chebyshev nodes, which is known to offer stable computations. Unfortunately, in many applications we often dispose

of equispaced samples, for which the Lebesgue constant grows exponentially. In this sense our fake nodes approach becomes essential.

We consider the domain $\Omega = [-5, 5]$ and both equispaced and randomly distributed nodes $\mathcal{X}_{n+1}$. Moreover, we evaluate the interpolants on a set of equispaced points $\Xi = \{\bar{x}_i, i = 0, \dots, 330\}$ and we compute the Relative Maximum Absolute Error (RMAE)

$$\text{RMAE} = \max_{i=0,\dots,m} \frac{|R^s_{n,f}(\bar{x}_i) - f(\bar{x}_i)|}{|f(\bar{x}_i)|}.$$

The experiment are carried out in PYTHON 3.6 using Numpy 1.15; see [20].

4.1. Applications to Runge phenomenon

Let us consider the Runge function

$$f_1(x) = \frac{1}{x^2 + 1}.$$

First, as samples, we consider equispaced interpolation nodes $\mathcal{E}_{n+1}$ on Ω and we compute and compare the following interpolation schemes:

i. the polynomial interpolating the equispaced points and associated function values $\mathcal{E}_{n+1}$ and $f_1(\mathcal{E}_{n+1})$, i.e the original data set and function values;

ii. the polynomial interpolating the CL nodes in Ω and resampled function values $\mathcal{C}_{n+1}$ and $f_1(\mathcal{C}_{n+1})$, i.e. we resample the function;

iii. the approximant built upon a polynomial interpolant on the fake CL nodes $S(\mathcal{E}_{n+1}) = \mathcal{C}_{n+1}$ and function values related to the equispaced points $f_1(\mathcal{E}_{n+1})$. In this setting, we use the map defined in (10).

In Fig. 1, we show different reconstructions of f_1 for a fixed number of nodes. We observe that the polynomial interpolation on equispaced nodes is affected by the Runge phenomenon, while the use of fake CL nodes performs even better than the one obtained via resampling on CL nodes. We investigate the asymptotic behaviors in Fig. 2: if resampling is feasible, then CL nodes asymptotically outperform all methods. Else, we observe that the reconstruction via fake nodes offers good approximations for small degrees and then stabilises. However, the main advantage is that it does not require resampling. Finally, in Fig. 3 we plot the Lebesgue functions related to the proposed methods. As pointed out in the theoretical analysis, the behavior of the fake nodes in terms of Lebesgue constant is analogous to classical polynomial interpolation on CL points.

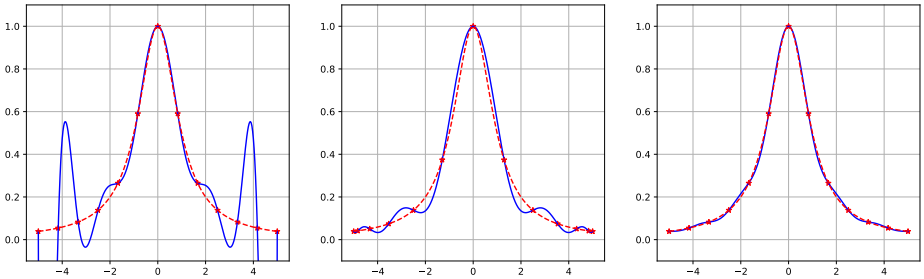

Figure 1: Interpolation with 13 points of the Runge function on $[-5, 5]$ using equispaced (left), CL (center) and fake nodes (right). The nodes are represented by stars, the original and reconstructed functions are plotted with continuous red and dotted blue lines, respectively.

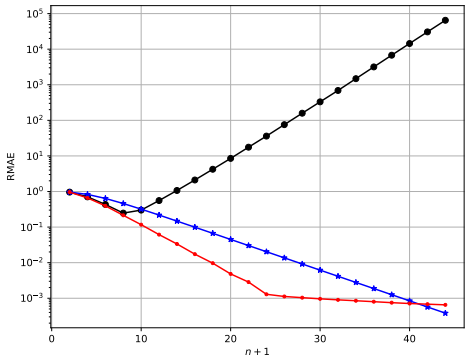

Figure 2: The RMAE for the Runge function varying the number of nodes. The results with equispaced, CL and fake nodes are represented by black circles, blue stars and red dots, respectively.

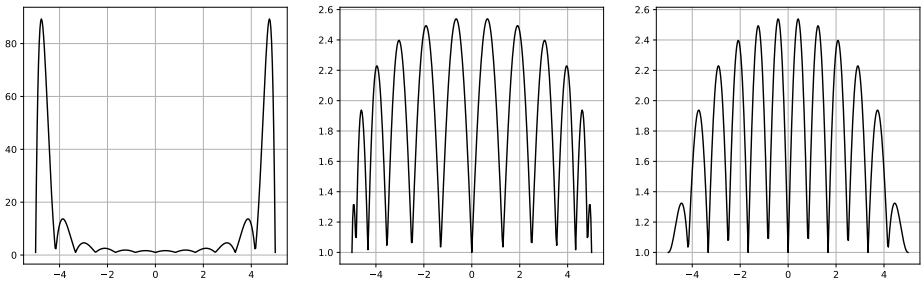

Figure 3: Lebesgue functions of equispaced (left), CL (center) and fake CL (right) nodes.

Finally, for a more general test, we consider a set of interpolation nodes $\mathcal{R}_{n+1} = \{r_i\}_{i=0,\dots,n}$ obtained by perturbing the internal nodes of $\mathcal{E}_{n+1} = \{e_i\}_{i=0,\dots,n}$ by adding a Gaussian noise, that is $r_i = e_i + \varepsilon_i$ for $i = 1, \dots, n-1$ with $\varepsilon_i \sim \mathcal{N}(\mu = 0, \sigma^2 = 0.04)$. We compute and compare:

i. the polynomial interpolating the random points and associated function values $\mathcal{R}_{n+1}$ and $f_1(\mathcal{R}_{n+1})$, i.e the original data set and function values;

ii. the approximant built upon a polynomial interpolant on the fake CL nodes $S(\mathcal{R}_{n+1}) = \mathcal{C}_{n+1}$ and function values related to the random points $f_1(\mathcal{R}_{n+1})$. In this setting, we use the map defined by Algorithm 1.

After generating the set $\mathcal{R}_{n+1}$ for $n = 20$, in Fig. 4 we show the reconstructions of f_1 obtained via standard polynomial interpolation on $\mathcal{R}_{n+1}$ and via the fake nodes approach. In the first case, the reconstruction is affected by the Runge phenomenon (RMAE$= 8.36\text{E}+01$), while the second method provides a stable interpolation process (RMAE$= 3.97\text{E}-02$). In Fig. 5 we plot the Lebesgue functions related to both methods.

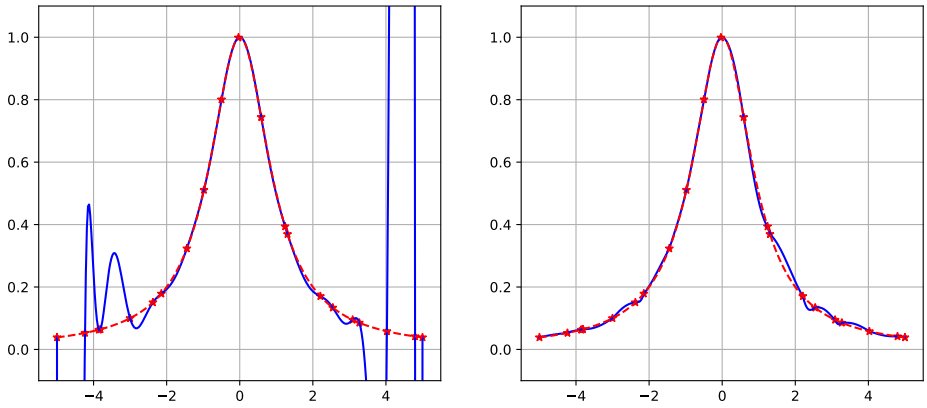

Figure 4: Interpolation with 21 random points with random seed 10 of the Runge function on $[-5, 5]$ using equispaced (left), and fake nodes (right). The nodes are represented by stars, the original and reconstructed functions are plotted with continuous red and dotted blue lines, respectively.

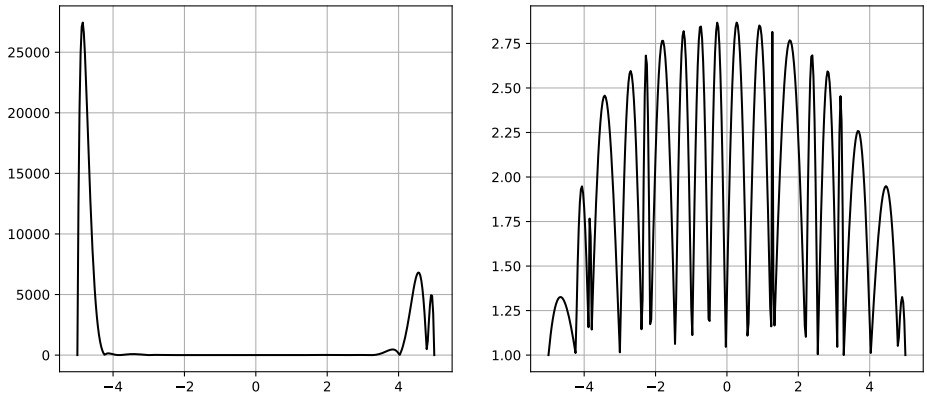

Figure 5: Lebesgue functions of random (left) and fake CL (right) nodes.

4.2. Applications to Gibbs phenomenon

In this subsection, we consider the problem of interpolating functions that present jump discontinuities. We define

$$f_2(x) \coloneqq \begin{cases} \dfrac{x^2}{10}, & -5 \leq x < -\frac{3}{2}, \\ \dfrac{1}{4}x + \dfrac{19}{8}, & -\frac{3}{2} \leq x < \frac{5}{2}, \\ -\dfrac{x^3}{30} + 4, & \frac{5}{2} \leq x \leq 5. \end{cases}$$

As done in the previous subsection, we can compare:

i. the polynomial interpolating the equispaced points and associated function values $\mathcal{E}_{n+1}$ and $f_2(\mathcal{E}_{n+1})$, i.e the original data set and function values;

ii. the polynomial interpolating the CL nodes in Ω and resampled function values $\mathcal{C}_{n+1}$ and $f_2(\mathcal{C}_{n+1})$, i.e. we resample the function;

iii. the approximant built upon a polynomial interpolant on the fake nodes $S(\mathcal{E}_{n+1})$ and function values related to the equispaced points $f_2(\mathcal{E}_{n+1})$. In this setting, we fix $k = 50$ and we use the map defined via Algorithm 2.

In Fig. 6 we display the results obtained using 20 interpolation points. We observe that the Gibbs phenomenon affects the reconstruction obtained via resampling on CL nodes, while it is mitigated using the proposed fake nodes approach. In Fig. 7 we show the related Lebesgue functions. Finally, in Fig. 8, we observe the asymptotic behavior of the discussed methods. The results are quite impressive, meaning that we are able to effectively deal with the Gibbs phenomenon by computing S via the simple idea of translating points (see Algorithm 2).

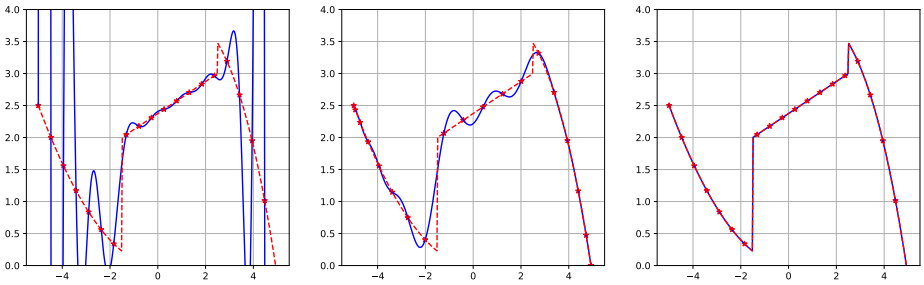

Figure 6: Interpolation with 20 points of the function f_2 on $[-5, 5]$, using equispaced (left), CL nodes (center) and the discontinuous map (right). The nodes are represented by stars, the original and reconstructed functions are plotted with continuous red and dotted blue lines, respectively.

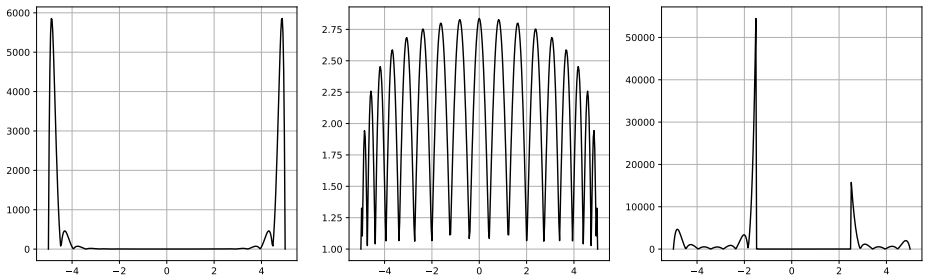

Figure 7: Lebesgue functions of equispaced (left), CL (center) and fake nodes (right).

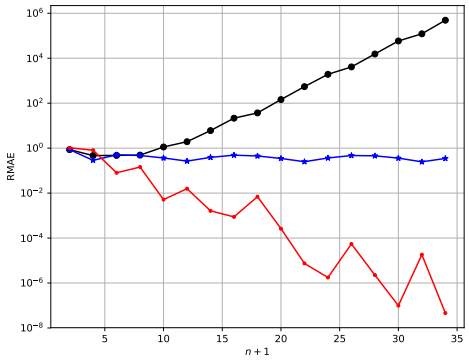

Figure 8: The RMAE for the function f_2 varying the number of nodes. The results with equispaced, CL and fake nodes are represented by black circles, blue stars and red dots, respectively.

5. Conclusions and future work

In this paper, we presented a very flexible tool that can be used as an effective alternative to classical polynomial interpolation. Its implementation turns out to be immediate and we also provided the PYTHON codes. The main advantage is that the approximation via mapped bases is carried out without resampling the function at the sought nodes. We also investigated two automated ways for defining the maps that lead to substantial improvements in the interpolation process.

Work in progress consists in extending the proposed procedure to other bases, such as trigonometric bases, with immediate applications to the reconstruction of (discontinuous) periodic signals, Floater-Hormann barycentric interpolation and splines; refer e.g. to [13, 14, 16]. Moreover, the extension to higher dimensions will be investigated in the context of radial basis functions.

Acknowledgments

We sincerely thank the reviewers for their insightful comments. 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.

Bibliography

[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] S. Amat, K. Dadourian, J. Liandrat, *On a nonlinear subdivision scheme avoiding Gibbs oscillations and converging towards C^s functions with $s > 1$*, Math. Comput. **80** (2011), 959–971.

[4] R. Archibald, A. Gelb, J. Yoon, *Fitting for edge detection in irregularly sampled signals and images*, SIAM J. Numer. Anal. **43** (2005), 259–279.

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

[6] J.P. Berrut, G. Elefante, *A periodic map for linear barycentric rational trigonometric interpolation*, preprint 2019.

[7] J.P. Berrut, H.D. Mittelmann, *Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval*, Comput. Math. Appl. **33** (1997), 77–86.

[8] 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.

[9] J.P. Boyd, F. Xu, *Divergence (Runge Phenomenon) for least-squares polynomial approximation on an equispaced grid and Mock–Chebyshev subset interpolation*, Appl. Math. and Comput. **210** (2009), 158–168.

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

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

[12] L. Brutman, *Lebesgue functions for polynomial interpolation – a survey*, Ann. Numer. Math. **4** (1997), 111–127.

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

[14] E. Cirillo, K. Hormann, J. Sidon, *Convergence rates of derivatives of Floater–Hormann interpolants for well-spaced nodes*, Appl. Num. Math. **116** (2017), 108–118.

[15] P.J. Davis, *Interpolation and Approximation*, Dover Publications, 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 of polynomial approximation methods on Lissajous curves with applications in MPI*, Dolomites Res. Notes Approx. **10** (2017), 128–137.

[19] S. De Marchi, F. Marchetti, E. Perracchione, *Jumping with Variably Scaled Discontinuos Kernels (VSDKs)*, preprint 2018.

[20] S. De Marchi, F. Marchetti, E, Perracchione, D. Poggiali, *Python code for Fake Nodes interpolation approach*, https://github.com/pog87/FakeNodes.

[21] B. Fornberg, N. Flyer, *The Gibbs Phenomenon in Various Representations and Applications*, chapter The Gibbs phenomenon for radial basis functions. Sampling Publishing, Potsdam, NY, 2008.

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

[23] N. Hale, L.N. Trefethen, *New quadrature formulas from conformal maps*, SIAM J. Numer. Anal. **46** (2008), 930–948.

[24] J.H. Jung, *A note on the Gibbs phenomenon with multiquadric radial basis functions*, Appl. Num. Math. **57** (2007), 213–219.

[25] 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.

[26] F. Piazzon, *Optimal polynomial admissible meshes on some classes of compact subsets of R^d*, J. Approx. Theory **207** (2016), 241–264.

[27] T.J. Rivlin, *An Introduction to the Approximation of Functions*, Dover Publications, New York, 1969.

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

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

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

[31] A.H. Turetskii, *The bounding of polynomials prescribed at equally distributed points*, Proc. Pedag. Inst. Vitebsk Vol. 3 (1940), 117–127 (in Russian).