Journal of Computational and Applied Mathematics 439 (2024) 115597

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A barycentric trigonometric Hermite interpolant via an iterative approach

Giacomo Elefante

Department of Mathematics "Tullio Levi-Civita", University of Padova, Via Trieste 63, Padova, Italy

ARTICLE INFO

Article history:
Received 26 April 2023
Received in revised form 24 July 2023

MSC:
65D05
65T40
42A15

Keywords:
Barycentric interpolation
Rational interpolation
Trigonometric interpolation
Hermite interpolation

ABSTRACT

In this work we construct an Hermite interpolant starting from basis functions that satisfy a Lagrange property. In fact, we extend and generalise an iterative approach, introduced by Cirillo and Hormann (2018) for the Floater–Hormann family of interpolants. Secondly, we apply this scheme to produce an effective barycentric rational trigonometric Hermite interpolant at general ordered nodes using as basis functions the ones of the trigonometric interpolant introduced by Berrut (1988). For an easy computational construction, we calculate analytically the differentiation matrix. Finally, we conclude with various examples and a numerical study of the convergence at equidistant nodes and conformally mapped nodes.

1. Introduction

Hermite interpolation is a classical problem which has been widely explored. It has many applications, and therefore, it is still vivid and studied.

The problem consists into finding a good function that satisfies the Hermite conditions, i.e.,

$$r_m^{(j)}(x_i) = f_{i,j} = f^{(j)}(x_i), \quad i = 0, \dots, n; \quad j = 0, \dots, m, \tag{1}$$

which means that it has to interpolate not only a function on some nodes x_i, $i = 0, \dots, n$ but also the values of its first m derivatives.

A good proposal for a solution of this problem could be given by interpolants based on rational functions since they may take advantage of the well-known stability of the latters for the classical problem of interpolation (see e.g. [1]). In fact, some recent works go in that direction, as the family of Hermite barycentric interpolants introduced in [2], the family of interpolants in [3] or the combination of rational functions and six-point Shepard basis functions introduced in [4].

A good stable interpolant that achieves a fast and accurate solution is preferable since it may be useful for the many applications where an Hermite interpolant is used, such as in Treecode algorithm [5] or constructing curves and surfaces for images [6] or approximating the reliability of a hammock network of arbitrary size [7].

Firstly, let us consider the iterative method, introduced in [3], to construct an Hermite interpolant starting from the basis of the renowned Floater–Hormann family of interpolants. This consists into considering primarily the basis functions $b_i(x)$ that are constructed, given $d \in \mathbb{N}$ and $x_0, \dots, x_n$ interpolation nodes, as

$$b_i(x) = \frac{w_i}{x - x_i} \Big/ \sum_{j=0}^{n} \frac{w_j}{x - x_k}, \; i = 0, \dots, n,$$

E-mail address: giacomo.elefante@unipd.it.

https://doi.org/10.1016/j.cam.2023.115597

with

$$w_i = (-1)^{i+d} \sum_{j=\max(0,i-d)}^{\min(i,n-d)} \prod_{k=j,k\neq i}^{j+d} \frac{1}{|x_i - x_k|}, \; i = 0, \ldots, n.$$

Therefore, once are considered the functions

$$b_{i,j}(x) = \frac{1}{j!}(x - x_i)^j b_i(x)^{j+1},$$

they can be used to construct an interpolant that satisfies the Hermite conditions.

The effectiveness of such construction for FH-interpolants has been investigated and in particular, in [8], it has been studied the Lebesgue constant at equidistant nodes and in [9], an estimate of the interpolation error has been given.

More specifically, we are going to firstly generalise this iterative Hermite construction and, later, use this method to introduce a barycentric rational trigonometric Hermite interpolant.

The paper is construct as follows: in Section 2 is presented the general iterative approach. In Section 3 is analysed the Hermite interpolant constructed by using as basis function the one of Berrut's trigonometric interpolant and, moreover, is computed analytically the differentiation matrix for the Hermite basis for general ordered nodes; this allows a fast construction and computation of the barycentric rational trigonometric Hermite interpolant. In Section 4 various numerical examples are presented and analysed. Finally, we conclude in Section 5.

2. A general iterative approach

Let us consider a set $\Omega \subset \mathbb{R}$ and some nodes $x_i \in \Omega$, $i = 0, \ldots, n$. Then, let us consider some basis functions b_i construct by using the nodes x_i's for an interpolant on Ω satisfying the Lagrange property $b_i(x_j) = \delta_{i,j}$. We define, therefore, the following functions,

$$b_{i,j}(x) := \frac{1}{j!} d_i(x)^j b_i(x)^{j+1}$$

for a smooth function $d_i(x)$ that vanishes in x_i and such that $d_i(x_j) \neq 0$, for $j \neq i$, and $d_i'(x_i) = 1$.

These functions will be the foundation to construct the Hermite interpolants, in fact we get the following.

Lemma 1. *Let $\mathcal{X}_n = \{x_0, \ldots, x_n\} \subset \Omega \subset \mathbb{R}$ be some ordered nodes and let $d_i(x)$ be a function that vanishes in x_j only if $j = i$ and such that $d_i'(x_i) = 1$. Let $b_i(x)$, $i = 0, \ldots, n$ be a basis of some space $\mathcal{V}$ that satisfy the Lagrange property at the nodes. Then, the functions $b_{i,j}$ defined as*

$$b_{i,j}(x) := \frac{1}{j!} d_i(x)^j b_i(x)^{j+1} \tag{2}$$

satisfy

$$b_{i,j}^{(k)}(x_\ell) = \begin{cases} 0, & \text{if } k < j, \\ \delta_{i,\ell}, & \text{if } k = j. \end{cases} \tag{3}$$

Proof. For $j = 0$ the statement follows directly from the Lagrange property of $b_i(x)$. Let us now consider $j > 0$ and prove the statement by induction on j.

Let

$$c_i(x) = d_i(x) b_i(x);$$

then, it is clear that

$$b_{i,j}(x) = \frac{1}{j} c_i(x) b_{i,j-1}(x).$$

Once we derive, we get by using the Leibniz rule that

$$b_{i,j}^{(k)}(x) = \frac{1}{j} \sum_{s=0}^{k} \binom{k}{s} c_i(x)^{(k-s)}(x) b_{i,j-1}^{(s)}(x).$$

Therefore, since $c_i(x_p) = 0$ for $p = 0, \ldots, n$ and, since for the induction hypothesis we have that

$$b_{i,j-1}^{(s)}(x_p) = 0 \text{ for } s < j - 1,$$

we get

$$b_{i,j}^{(k)}(x_p) = \frac{1}{j} \sum_{s=j-1}^{k-1} \binom{k}{s} c_i^{(k-s)}(x_p) b_{i,j-1}^{(s)}(x_p).$$

Let us observe that the sum is empty in the case when $k < j$, whereas when $k = j$ we obtain

$$b_{i,j}^{(j)}(x_p) = \frac{1}{j}\binom{j}{j-1} c_i'(x_p) b_{i,j-1}^{(j-1)}(x_p).$$

Finally, due to the induction hypothesis on $b_{i,j-1}^{(j-1)}(x_p)$, we notice that $b_{i,j}^{(j)}(x_p) = 0$ if $p \neq i$ and when $p = i$ we have that the product is one since $c_i'(x_i) = 1$ by construction. □

Remark 1. Given a function $h_i(x)$ vanishing only in x_i and with a non-zero derivative in x_i, we remark that we can always construct a function that satisfies the assumption of Lemma 1 by setting

$$d_i(x) := \frac{h_i(x)}{h_i'(x_i)}.$$

Once we have the functions $b_{i,j}$ we can construct the Hermite interpolant starting from

$$r_0(x) = \sum_{i=0}^{n} b_{i,0}(x) f_{i,0},$$

which corresponds to the classical interpolant since $b_{i,0} = b_i$. For $j > 0$, since we have that normally

$$r_{j-1}^{(j)}(x_\ell) \neq f_{\ell,j},$$

we introduce, at each step, a correction function q_j which we add to the previous interpolants, in order to construct a new one that interpolates correctly the derivatives up to the jth, i.e.,

$$r_j(x) = r_{j-1}(x) + q_j(x), \quad j = 1, \ldots, m. \tag{4}$$

Therefore, we define the correction term as

$$q_j(x) = \sum_{i=0}^{n} b_{i,j}(x)\left(f_{i,j} - r_{j-1}^{(j)}(x_i)\right), \quad j = 1, \ldots, m. \tag{5}$$

The resulting function r_m, which we may write shortly as

$$r_m(x) := \sum_{j=0}^{m}\sum_{i=0}^{n} b_{i,j}(x) g_{i,j}$$

for proper $g_{i,j}$, construct via the iterative approach described above, thanks to Lemma 1, will satisfy the Hermite conditions. In fact, we have the following theorem.

Theorem 1. *Let $\mathcal{X}_n = \{x_0, \ldots, x_n\} \subset \Omega \subset \mathbb{R}$ be some ordered nodes and let $d_i(x)$ be a function that vanishes in x_j only if $j = i$ and such that $d_i'(x_i) = 1$. Let $b_i(x)$, $i = 0, \ldots, n$ be a basis of some space $\mathcal{V}$ that satisfy the Lagrange property at the nodes and let $b_{i,j}$ be defined as (2). Then, the function r_m, defined as*

$$r_m(x) := \sum_{j=0}^{m}\sum_{i=0}^{n} b_{i,j}(x) g_{i,j}, \tag{6}$$

with

$$g_{i,j} = \begin{cases} f_{i,0} & \text{if } j = 0 \\ f_{i,j} - r_{j-1}^{(j)}(x_i) & \text{if } j > 0, \end{cases} \tag{7}$$

satisfies the Hermite conditions (1)

Proof. It is straightforward that,

$$r_m^{(k)}(x_\ell) = \sum_{i=0}^{n}\sum_{j=0}^{m} b_{i,j}^{(k)}(x)_{|_{x=x_\ell}} g_{i,j},$$

which due to Lemma 1 is

$$\begin{aligned} r_m^{(k)}(x_\ell) &= \sum_{i=0}^{n}\sum_{j=0}^{k} b_{i,j}^{(k)}(x)_{|_{x=x_\ell}} g_{i,j} \\ &= g_{\ell,k} + \sum_{i=0}^{n}\sum_{j=0}^{k-1} b_{i,j}^{(k)}(x)_{|_{x=x_\ell}} g_{i,j} \end{aligned}$$

$$= g_{\ell,k} + r_{k-1}^{(k)}(x_\ell)$$
$$= f_{\ell,k}. \quad \square$$

3. A barycentric rational trigonometric interpolant

Barycentric rational trigonometric interpolants are suitable for approximating periodic functions and many of them have been introduced in the last years.

The barycentric form of the classical interpolant at equidistant nodes has been introduced by Henrici [10], and, later, Berrut [11] proposed to use the same function as the Henrici's interpolant but for other ordered nodes.

More recently, Bandiziol and De Marchi [12] proposed a barycentric rational trigonometric interpolant constructed similarly as the well-known Floater–Hormann interpolant. Moreover, Baddoo [13] introduced a trigonometric equivalent of the renowned algorithm AAA (see [14]), the AAAtrig, which construct a barycentric trigonometric rational approximant selecting the nodes progressively via a greedy algorithm.

In this section, we focus into Berrut's trigonometric interpolant [11]. Let us consider n ordered nodes $0 \le \theta_0 < \cdots < \theta_{n-1} < 2\pi$, then, we define the interpolant as

$$T_n(\theta) = \frac{\sum_{i=0}^{n-1}(-1)^i \operatorname{cst}\left(\frac{\theta-\theta_i}{2}\right) f(\theta_i)}{\sum_{i=0}^{n-1}(-1)^i \operatorname{cst}\left(\frac{\theta-\theta_i}{2}\right)}, \tag{8}$$

where the function cst is

$$\operatorname{cst}(\theta) = \begin{cases} \csc(\theta), & \text{if } n \text{ is odd,} \\ \operatorname{ctg}(\theta), & \text{if } n \text{ is even.} \end{cases} \tag{9}$$

It has no poles in $[0, 2\pi)$ and enjoys a logarithmic growth of its Lebesgue constant for a wide family of interpolation nodes [15]. This class of nodes includes also the images of equidistant nodes via a conformal map, such as those in [16,17]. As said earlier, it corresponds to the classical trigonometric interpolant when the nodes are equidistant [10] and, therefore, it enjoys its properties, such as the exponential convergence. In addition, the interpolant converges exponentially also when the nodes are images of equidistant nodes via a conformal map [18] and it has been used effectively in [19] to interpolate functions on two-dimensional starlike domains.

In order to construct the Hermite interpolant and to retain the periodic behaviour, we may choose

$$d_i(\theta) = 2\sin\left(\frac{\theta-\theta_i}{2}\right), \tag{10}$$

which clearly satisfies the conditions we need to define the functions $b_{i,j}$.

In particular, we get

$$\begin{aligned} b_{i,j}(\theta) &:= \frac{2^j}{j!}\sin^j\left(\frac{\theta-\theta_i}{2}\right)(-1)^{ij+i}\frac{\operatorname{cst}^{j+1}\left(\frac{\theta-\theta_i}{2}\right)}{\left(\sum_{k=0}^{n-1}(-1)^k \operatorname{cst}\left(\frac{\theta-\theta_k}{2}\right)\right)^{j+1}} \\ &= \frac{2^j}{j!}(-1)^{ij+i}\frac{\operatorname{cst}\left(\frac{\theta-\theta_i}{2}\right)}{\left(\sum_{k=0}^{n-1}(-1)^k \operatorname{cst}\left(\frac{\theta-\theta_k}{2}\right)\right)^{j+1}} \cdot \begin{cases} 1, & \text{if } n \text{ is odd,} \\ \cos^j\left(\frac{\theta-\theta_i}{2}\right), & \text{if } n \text{ is even,} \end{cases} \\ &= \frac{2^j(-1)^{i(j+1)}}{j!}\operatorname{cst}\left(\frac{\theta-\theta_i}{2}\right) \begin{cases} \dfrac{1}{\left(\sum_k \frac{(-1)^k}{\sin\left((\theta-\theta_k)/2\right)}\right)^{j+1}} & \text{if } n \text{ is odd,} \\ \dfrac{\cos^j\left(\frac{\theta-\theta_i}{2}\right)}{\left(\sum_k \frac{(-1)^k}{\sin\left((\theta-\theta_k)/2\right)}\right)^{j+1}} & \text{if } n \text{ is even,} \end{cases} \\ &= \frac{\frac{2^j(-1)^{i(j+1)}}{j!}\operatorname{cst}\left(\frac{\theta-\theta_i}{2}\right)\eta_{i,j}^{(n)}(\theta),}{\left(\sum_{k=0}^{n-1}\frac{(-1)^k}{\sin((\theta-\theta_k)/2)}\right)^{j+1}} \end{aligned} \tag{11}$$

with

$$\eta_{i,j}^{(n)}(\theta) = \begin{cases} 1 & \text{if } n \text{ is odd,} \\ \cos^j\left(\frac{\theta-\theta_i}{2}\right) & \text{if } n \text{ is even.} \end{cases}$$

This formulation will allow us to calculate analytically the differential matrix of the Hermite interpolant

$$t_m(\theta) = \sum_{i=0}^{n-1}\sum_{j=0}^{m} b_{i,j}(\theta) g_{i,j}, \tag{12}$$

which will be useful for a fast computation of the interpolant.

Theorem 2. *Let us consider the basis functions (11), then when $s > j$ the differential matrix, whose elements are $(D_j^s)_{ik} = b_{k,j}^{(s)}(\theta_i)$, is defined as*

$$\begin{aligned}(D_j^s)_{ik} =& \frac{1}{\sin\left(\frac{\theta_i-\theta_k}{2}\right)}\left((-1)^{(k-i)(j+1)}\sum_{q=0}^{\lfloor\frac{s-j-1}{2}\rfloor}\binom{s}{2q+1}\frac{(-1)^q}{2^{2q+1}}(D_j^{s-2q-1})_{ii}\right.\\ &\left.-\sum_{q=j}^{s-1}\binom{s}{s-q}\left(\sin\left(\frac{\theta-\theta_k}{2}\right)\right)^{(s-q)}_{|\theta=\theta_i}(D_j^q)_{ik}\right)\end{aligned} \tag{13}$$

when n is odd and

$$\begin{aligned}(D_j^s)_{ik} =& \frac{1}{\tan\left(\frac{\theta_i-\theta_k}{2}\right)}\left((-1)^{(k-i)(j+1)}\sum_{q=0}^{\lfloor\frac{s-j}{2}\rfloor}\binom{s}{2q+1}2^{-2q-1}\cdot\right.\\ &\cdot\left(\sum_{j=0}^{q}a(2q+1,2j+1)\right)(D_j^{s-2q-1})_{ii}\\ &\left.-\sum_{q=j}^{s-1}\binom{s}{s-q}\left(\tan\left(\frac{\theta-\theta_k}{2}\right)\right)^{(s-q)}_{|\theta=\theta_i}(D_j^q)_{ik}\right).\end{aligned} \tag{14}$$

when n is even, with

$$a(p,q) = (-1)^{\frac{1}{2}(p-\frac{3+(-1)^p}{2})}\,2\sum_{j=0}^{\frac{p-q-1}{2}}(-1)^{\frac{p-q-1}{2}-j}\binom{p+1}{j}\left(\frac{p-q-1}{2}-j+1\right)^p.$$

Proof. Therefore, we can write the basis as

$$b_{k,j}(\theta) := u_{k,j}\alpha(\theta)\,\mathrm{cst}\left(\frac{\theta-\theta_k}{2}\right) \tag{15}$$

where

$$u_{k,j} := \frac{2^j(-1)^{k(j+1)}}{j!} \tag{16}$$

and, if n is odd,

$$\alpha(\theta) := \frac{1}{\left(\sum_{i=0}^{n-1}(-1)^i\,\mathrm{cst}((\theta-\theta_i)/2)\right)^{j+1}}$$

and, if n is even,

$$\alpha(\theta) := \frac{\cos^j\left(\frac{\theta-\theta_k}{2}\right)}{\left(\sum_{i=0}^{n-1}(-1)^i\,\mathrm{cst}\,((\theta-\theta_i)/2)\right)^{j+1}}.$$

In this way we can compute the differentiation matrices as done for the classical barycentric trigonometric interpolant in [18], which will be useful to compute the values of the derivatives of the interpolant r_m at the nodes in the previous iterations.

If n is odd, considering Eq. (15), differentiating both side and evaluating at the points x_i, we get that

$$\begin{aligned}\left(\alpha(\theta)\right)^{(s)}_{|\theta=\theta_k} &= \frac{1}{u_{k,j}}\sum_{q=0}^{s}\binom{s}{q}\left(\sin\left(\frac{\theta-\theta_k}{2}\right)\right)^{(q)}|_{\theta=\theta_k}\left(b_{k,j}(\theta)\right)^{(s-q)}|_{\theta=\theta_k}\\ &= \frac{1}{u_{k,j}}\sum_{q=0}^{s}\binom{s}{q}\left(\sin\left(\frac{\theta-\theta_k}{2}\right)\right)^{(q)}|_{\theta=\theta_k}(D_j^{s-q})_{kk}\end{aligned}$$

$$= \frac{1}{u_{k,j}} \sum_{q=1}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_k} (D_j^{s-q})_{kk}, \tag{17}$$

where we defined $(D_j^s)_{ik} = b_{k,j}^{(s)}(\theta_i)$.

Moreover, by evaluating the derivative in different nodes we have

$$\begin{aligned} \big(\alpha(\theta)\big)^{(s)}\big|_{\theta=\theta_i} &= \frac{1}{u_{k,j}} \sum_{q=0}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} \big(b_{k,j}(\theta)\big)^{(s-q)}\big|_{\theta=\theta_i} \\ &= \frac{1}{u_{k,j}} \sum_{q=0}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ik}. \end{aligned} \tag{18}$$

In (18) we can isolate the term $q = 0$ of the sum, that gives,

$$\begin{aligned} \big(\alpha(\theta)\big)^{(s)}\big|_{\theta=\theta_i} =& \frac{1}{u_{k,j}} \sum_{q=1}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ik} \\ &+ \frac{1}{u_{k,j}} \left(\sin\left(\frac{\theta_i - \theta_k}{2} \right) \right) (D_j^s)_{ik}, \end{aligned} \tag{19}$$

which together with (17) gives us the following recursive formula for the differentiation matrix

$$\begin{aligned} (D_j^s)_{ik} = & \frac{1}{\sin\left(\frac{\theta_i - \theta_k}{2}\right)} \Bigg(u_{k,j} \big(\alpha(\theta)\big)^{(s)}\big|_{\theta=\theta_i} \\ & - \sum_{q=1}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ik} \Bigg) \end{aligned} \tag{20}$$

$$\begin{aligned} = & \frac{1}{\sin\left(\frac{\theta_i - \theta_k}{2}\right)} \Bigg(\frac{u_{k,j}}{u_{i,j}} \sum_{q=1}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_i}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ii} \\ & - \sum_{q=1}^{s} \binom{s}{q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ik} \Bigg). \end{aligned} \tag{21}$$

Notice that since the terms $b_{i,j}$ satisfy (3), D_j^j corresponds to the identity matrix and $D_j^s = 0$ when $s < j$. Then, the formula becomes

$$\begin{aligned} (D_j^s)_{ik} = & \frac{1}{\sin\left(\frac{\theta_i - \theta_k}{2}\right)} \Bigg(\frac{u_{k,j}}{u_{i,j}} \sum_{q=j}^{s-1} \binom{s}{s-q} \left(\sin\left(\frac{\theta - \theta_i}{2} \right) \right)^{(s-q)} \Big|_{\theta=\theta_i} (D_j^q)_{ii} \\ & - \sum_{q=j}^{s-1} \binom{s}{s-q} \left(\sin\left(\frac{\theta - \theta_k}{2} \right) \right)^{(s-q)} \Big|_{\theta=\theta_i} (D_j^q)_{ik} \Bigg). \end{aligned} \tag{22}$$

Similarly, when n is even, we have that

$$\big(\alpha(\theta)\big)^{(s)}\big|_{\theta=\theta_k} = \frac{1}{u_{k,j}} \sum_{q=1}^{s} \binom{s}{q} \left(\tan\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_k} (D_j^{s-q})_{kk} \tag{23}$$

$$\big(\alpha(\theta)\big)^{(s)}\big|_{\theta=\theta_i} = \frac{1}{u_{k,j}} \sum_{q=0}^{s} \binom{s}{q} \left(\tan\left(\frac{\theta - \theta_k}{2} \right) \right)^{(q)} \Big|_{\theta=\theta_i} (D_j^{s-q})_{ik}. \tag{24}$$

Hence,

$$\begin{aligned} (D_j^s)_{ik} = & \frac{1}{\tan\left(\frac{\theta_i - \theta_k}{2}\right)} \Bigg(\frac{u_{k,j}}{u_{i,j}} \sum_{q=j}^{s-1} \binom{s}{s-q} \left(\tan\left(\frac{\theta - \theta_i}{2} \right) \right)^{(s-q)} \Big|_{\theta=\theta_i} (D_j^q)_{ii} \\ & - \sum_{q=j}^{s-1} \binom{s}{s-q} \left(\tan\left(\frac{\theta - \theta_k}{2} \right) \right)^{(s-q)} \Big|_{\theta=\theta_i} (D_j^q)_{ik} \Bigg). \end{aligned} \tag{25}$$

Moreover, since

$$\left(\sin\left(\frac{\theta-\theta_k}{2}\right)\right)^{(q)} = \begin{cases} \frac{(-1)^{\frac{q}{2}}}{2^q}\sin\left(\frac{\theta-\theta_k}{2}\right), & \text{if } q \text{ even} \\ \frac{(-1)^{\frac{q-1}{2}}}{2^q}\cos\left(\frac{\theta-\theta_k}{2}\right), & \text{if } q \text{ odd} \end{cases}$$

and since the jth derivative of $\tan((\theta-\theta_i)/2)$ computed in θ_i is a given by (see [20])

$$\left(\tan\left(\frac{\theta-\theta_i}{2}\right)\right)^{(n)}\Big|_{\theta=\theta_i} = \begin{cases} 0, & \text{if } n=2k, \\ 2^{-n}\sum_{j=0}^{k-1} a(2k-1,2j+1), & \text{if } n=2k-1, \end{cases}$$

for some $k \in \mathbb{N}$ and with

$$a(p,q) = (-1)^{\frac{1}{2}(p-\frac{3+(-1)^p}{2})}\, 2 \sum_{j=0}^{\frac{p-q-1}{2}} (-1)^{\frac{p-q-1}{2}-j}\binom{p+1}{j}\left(\frac{p-q-1}{2}-j+1\right)^p$$

we can simplify Eq. (22) to

$$\begin{aligned}(D_j^s)_{ik} = \frac{1}{\sin\left(\frac{\theta_i-\theta_k}{2}\right)}\Bigg(& (-1)^{(k-i)(j+1)} \sum_{q=0}^{\lfloor\frac{s-j-1}{2}\rfloor}\binom{s}{2q+1}\frac{(-1)^q}{2^{2q+1}}(D_j^{s-2q-1})_{ii} \\ & - \sum_{q=j}^{s-1}\binom{s}{s-q}\left(\sin\left(\frac{\theta-\theta_k}{2}\right)\right)^{(s-q)}_{|_{\theta=\theta_i}}(D_j^q)_{ik}\Bigg)\end{aligned} \tag{26}$$

and Eq. (25) to

$$\begin{aligned}(D_j^s)_{ik} = \frac{1}{\tan\left(\frac{\theta_i-\theta_k}{2}\right)}\Bigg(& (-1)^{(k-i)(j+1)} \sum_{q=0}^{\lfloor\frac{s-j}{2}\rfloor}\binom{s}{2q+1}2^{-2q-1}\cdot \\ & \cdot\left(\sum_{j=0}^{q} a(2q+1,2j+1)\right)(D_j^{s-2q-1})_{ii} \\ & - \sum_{q=j}^{s-1}\binom{s}{s-q}\left(\tan\left(\frac{\theta-\theta_k}{2}\right)\right)^{(s-q)}_{|_{\theta=\theta_i}}(D_j^q)_{ik}\Bigg). \quad \square\end{aligned} \tag{27}$$

The interpolant (12) then can be written in the following extended form

$$t_m(\theta) = \sum_{i=0}^{n-1}\sum_{j=0}^{m} b_{i,j}(\theta)\left(f_{i,j} - \sum_{s=0}^{j-1}\sum_{k=0}^{n-1}(D_s^j)_{ik}f_{k,s}\right) \tag{28}$$

or the following form linear on the data

$$t_m(\theta) = \sum_{i=0}^{n-1}\sum_{j=0}^{m}\left(b_{i,j}(\theta) - \sum_{s=j+1}^{m}\sum_{k=0}^{n-1} b_{k,s}(\theta)(D_j^s)_{ki}\right)f_{i,j}, \tag{29}$$

which is specifically

$$\begin{aligned}t_m(\theta) = \Bigg(& \sum_{i=0}^{n-1}\sum_{j=0}^{m}\Bigg(u_{i,j}\,\mathrm{cst}\left(\frac{\theta-\theta_i}{2}\right)\eta_{i,j}^{(n)}(\theta)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-j} \\ & - \sum_{s=j+1}^{m}\sum_{k=0}^{n-1} u_{k,s}\,\mathrm{cst}\left(\frac{\theta-\theta_k}{2}\right)\eta_{i,j}^{(n)}(\theta)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-s}(D_j^s)_{ki}\Bigg)f_{i,j}\Bigg) \\ & \Big/\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m+1},\end{aligned} \tag{30}$$

with $u_{i,j}$ defined as in (16).

Once we consider the interpolant of the function $f \equiv 1$, the interpolant resolves to

$$t_m(\theta) = \left(\sum_{i=0}^{n-1}\sum_{j=0}^{m}\left(\sum_{k=0}^{n-1} u_{k,j}(D_0^j)_{ki}\right)\mathrm{cst}\left(\frac{\theta-\theta_i}{2}\right)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-j}\right) \Bigg/ \left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m+1}, \tag{31}$$

since the derivatives of f are zeros, and D_0^0 is the identity matrix.

Therefore we get the barycentric form of the interpolant, that is

$$t_m(\theta) = \left(\sum_{i=0}^{n-1}\sum_{j=0}^{m}\left(u_{i,j}\,\mathrm{cst}\left(\frac{\theta-\theta_i}{2}\right)\eta_{i,j}^{(n)}(\theta)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-j} - \sum_{s=j+1}^{m}\sum_{k=0}^{n-1} u_{k,s}\,\mathrm{cst}\left(\frac{\theta-\theta_k}{2}\right)\eta_{i,j}^{(n)}(\theta)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-s}(D_j^s)_{ki}\right) f_{i,j}\right) \Bigg/ \left(\sum_{i=0}^{n-1}\sum_{j=0}^{m}\left(\sum_{k=0}^{n-1} u_{k,j}(D_0^j)_{ki}\right)\mathrm{cst}\left(\frac{\theta-\theta_i}{2}\right)\left(\sum_{i=0}^{n-1}\frac{(-1)^i}{\sin\left(\frac{\theta-\theta_i}{2}\right)}\right)^{m-j}\right) \tag{32}$$

In addition, we remark that, when we compute the matrix for $s = j+1$, the elements simplify to the following

$$(D_j^{j+1})_{ik} = \frac{(-1)^{(j+1)(k-i)}(j+1)}{2}\,\mathrm{cst}\left(\frac{\theta_i-\theta_k}{2}\right), \qquad k \neq i, \tag{33}$$

which for $j = 0$ corresponds to the values in [18]; for the diagonal elements, we use the relation $\sum_{i=0}^{n-1} b_{i,j}^{(j)}(\theta) = 1$, since we want to reproduce the constants for the jth derivate. This implies that

$$\sum_{i=0}^{n-1} b_{i,j}^{(j+1)}(\theta) = 0,$$

so that we obtain

$$(D_j^{j+1})_{ii} = -\sum_{\substack{k=0\\k\neq i}}^{n-1}(D_j^{j+1})_{ik}.$$

Moreover, since for the construction of the Hermite interpolant (12), we need to compute $r_{j-1}^{(s)}(x_i)$ and therefore D_j^s, in the next section, we decided to proceed by computing the differentiation matrices of higher order as

$$D_j^s = (D_j^{j+1})^{s-j} \qquad s \geq j+1. \tag{34}$$

as done for the polynomial case in [21].

4. Numerical experiments

Here we present some numerical tests, some Matlab demos to reproduce the experiments are available at https://github.com/gelefant/TrigonometricHermite

We considered the following periodic test functions in $[0, 2\pi)$,

$$f_1(\theta) = e^{2\sin(\theta)+\cos(\theta)},$$
$$f_2(\theta) = \cos(3\theta) + \log(\cos(\theta)+1.5),$$

and we interpolated them by means of the interpolant (12) with the basis presented in Section 3 with N equidistant nodes.

We tested the interpolants which satisfy the Hermite conditions from one to four derivatives. In Fig. 1 we display both test functions together with two interpolants, one satisfying the Hermite conditions up to the second derivative and one satisfying the conditions up to the third derivative.

For a fixed m, we compute, the absolute error between the function and the interpolant with N nodes $\mathrm{err}_N = \max_{x\in[0,2\pi)} |f(x) - t_m(x)|$ and we can notice that the interpolant, in the case of conditions up to the second derivative

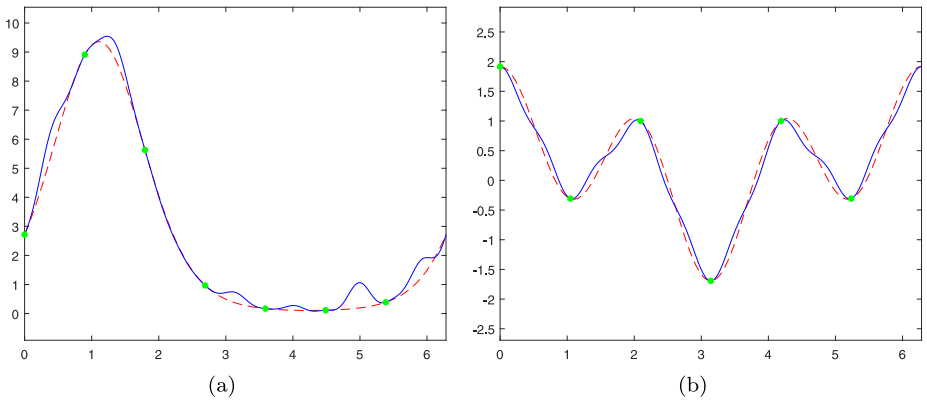


Fig. 1. The functions f_1 (left) and f_2 (right) in red dashed and in blue line the interpolant t_3 at 7 equidistant nodes (left) and the interpolant t_2 at 6 equidistant nodes (right). Interpolation nodes are labelled in green.

Table 1
Computation of the values $-\log_2(\frac{\text{err}_{2n}}{\text{err}_n})$ for t_3 and t_4 with both test functions.

	n	5	10	20	40	80	160
t_3	f_1	3.70	2.59	3.03	2.97	3.00	3.00
	f_2	4.18	2.96	2.89	2.97	3.01	2.99
t_4	f_1	4.37	2.93	2.93	2.95	2.87	2.96
	f_2	5.50	2.84	2.94	2.99	2.92	2.89

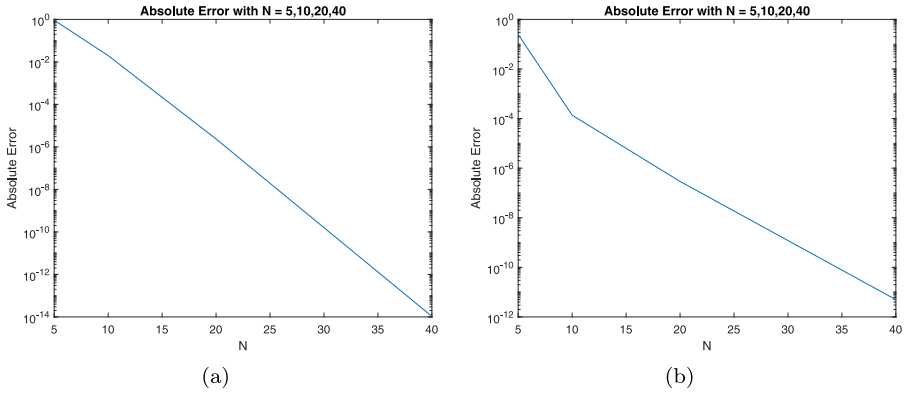


Fig. 2. Convergence of t_1 for f_1 (left) and f_2 (right) and $N = 5, 10, 20, 40$.

retains the exponential convergence of the classical interpolant (8) at equidistant nodes (see Figs. 2 and 3), on the other hand, when we have conditions on the third or the fourth derivatives, the convergence slows down (see Figs. 4 and 5). We can numerically estimate the convergence of the interpolant with N nodes as $\mathcal{O}(N^{-3})$. In fact, once we compute $-\log_2(\frac{\text{err}_{2n}}{\text{err}_n})$ with various n, we have a numerical estimate of the rate of convergence and it appears, as we can see in Table 1 where we computed it for $n = 5, 10, 20, 40, 80, 160$, that this ratio is almost 3.

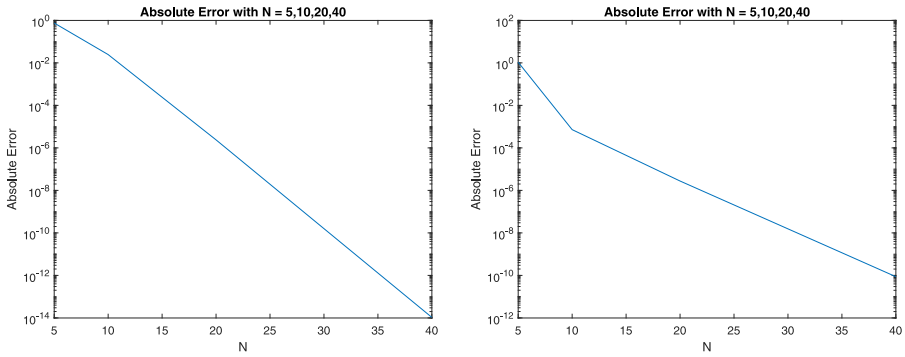
In order to test the interpolant with conformally shifted nodes and explore the convergence to functions with fronts, we tested the interpolant moreover with the function

$$f_3(\theta) = \tanh(50\cos(\theta + \pi/3));$$

which has two fronts located in $\theta_1 = \pi/6$ and $\theta_2 = 7\pi/6$. For this reason we compared the performances of the interpolant with equidistant nodes and conformally shifted nodes, via the conformal map introduced in [17], i.e.,

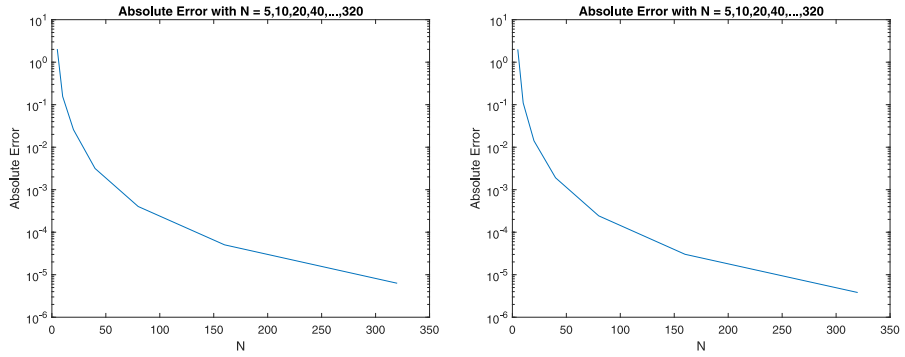
$$g_{\alpha,\beta}(\theta) = -\iota \log\left(\frac{e^{\iota\theta} + \alpha e^{\iota\beta}}{1 + \alpha e^{\iota(\theta-\beta)}}\right),$$

and using the generalisation for two fronts, where we set the same density parameter for both fronts as $\alpha = 0.85$.



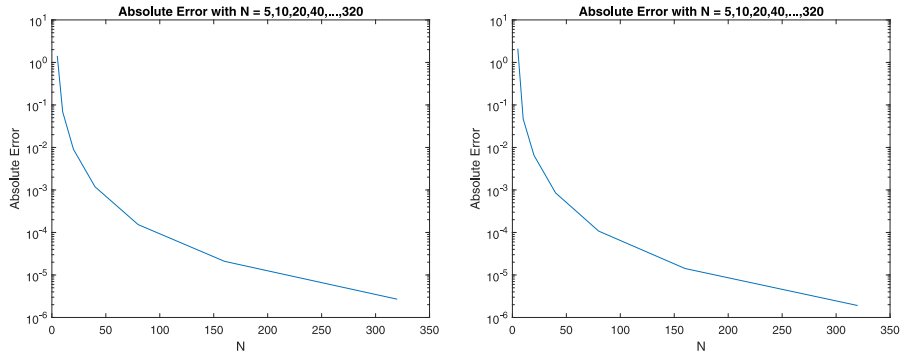
(a) (b)

Fig. 3. Convergence of t_2 for f_1 (left) and f_2 (right) and $N = 5, 10, 20, 40$.



(a) (b)

Fig. 4. Convergence of t_3 for f_1 (left) and f_2 (right) and $N = 5, 10, 20, 40, 80, 160, 320$.



(a) (b)

Fig. 5. Convergence of t_4 for f_1 (left) and f_2 (right) and $N = 5, 10, 20, 40, 80, 160, 320$.

As we can see in Figs. 6, the advantage of clustering the nodes in the locations of the fronts becomes less important as we consider more derivatives. In fact, it speeds up a lot the convergence in the case of one derivative but, then, once we add the conditions on more derivatives, clustering the nodes decreases the speed up until it even slows the convergence when we have conditions up to the fourth derivative.

We present another numerical experiment which is the reconstruction of a closed simple curve, given some points, as might be a figure or a sketch. The two dimensions are reconstructed separately and each coordinates is consider as a periodic function. Therefore, in order to enjoy the fast exponential convergence, the data are considered as the values of a function at equidistant nodes. Then, we approximate the first derivatives by using the coordinates of the difference quotients of the points.

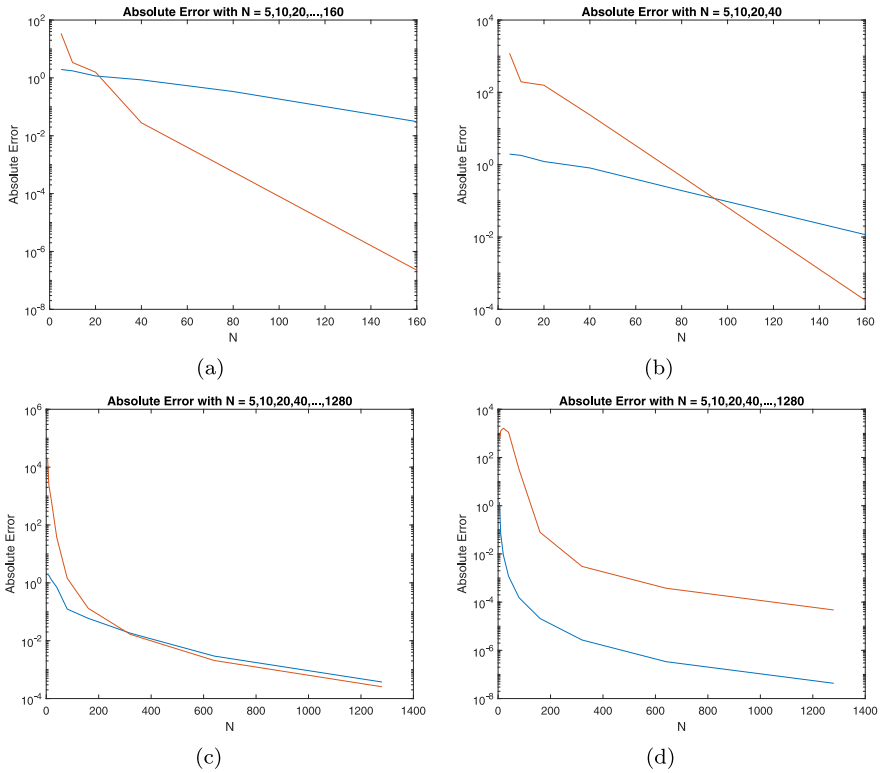


Fig. 6. Error of function f_3 with equidistant nodes (blue line) and conformally shifted nodes (red line) by using (a) t_1, (b) t_2, (c) t_3, (d) t_4, with $N = 5, 10, 20, \ldots, 160$ in (a) and (b) and $N = 5, 10, 20, \ldots, 1280$ in (c) and (d).

We did therefore, the reconstruction of the sketch in Fig. 7a and starting by extracting 362 points (which are the red dots in 7b), we used the values of the abscissae and ordinates as datae of two periodic function with those values at equidistant nodes and which we intend to approximate by the Hermite interpolant t_1, which is the interpolant with conditions on one derivative. Moreover, since the datae of the real tangent vectors are unknown, we approximate them with the difference quotient vector between the extracted points ordered in anticlockwise sense. The result is in Fig. 7b.

In Fig. 7(c)–(e) we used a subset of the initial set of nodes: half points, a third and a fifth, respectively.

As we can see by the results in Figs. 7, the interpolant performs quite well even by using an approximation of the derivative and by decreasing the number of nodes.

In particular, even by using a fifth of the points initially extracted the reconstruction is still resembling quite a lot the sketch.

5. Conclusion

The present work introduced an iterative scheme to construct an Hermite interpolant starting from a general interpolation basis and in particular, we focused on Berrut's trigonometric interpolant. In order to construct the values of the derivative of the precedent iterative step, we compute analytically the differentiation matrix which allows to obtain these values easily as a matrix product.

Our numerical experiments suggest that the produced interpolant retains the exponential convergence of the classical interpolant at equidistant nodes and conformally shifted nodes, once we interpolate the first two derivatives and slows down to a rate of $\mathcal{O}(N^{-3})$, where N is the number of nodes used, when we interpolate also the third or the fourth derivatives.

Moreover, the tests suggest that clustering nodes in the locations of fronts via a conformal map speeds up the convergence in case of one derivative but, instead, could be of negative impact when we need to satisfy conditions also for higher derivatives.

Future work will be aimed into analysing the convergence more deeply, also for other class of nodes, as the one introduced in [17], which includes conformally shifted nodes, and also into analysing the Lebesgue constant of the interpolant.

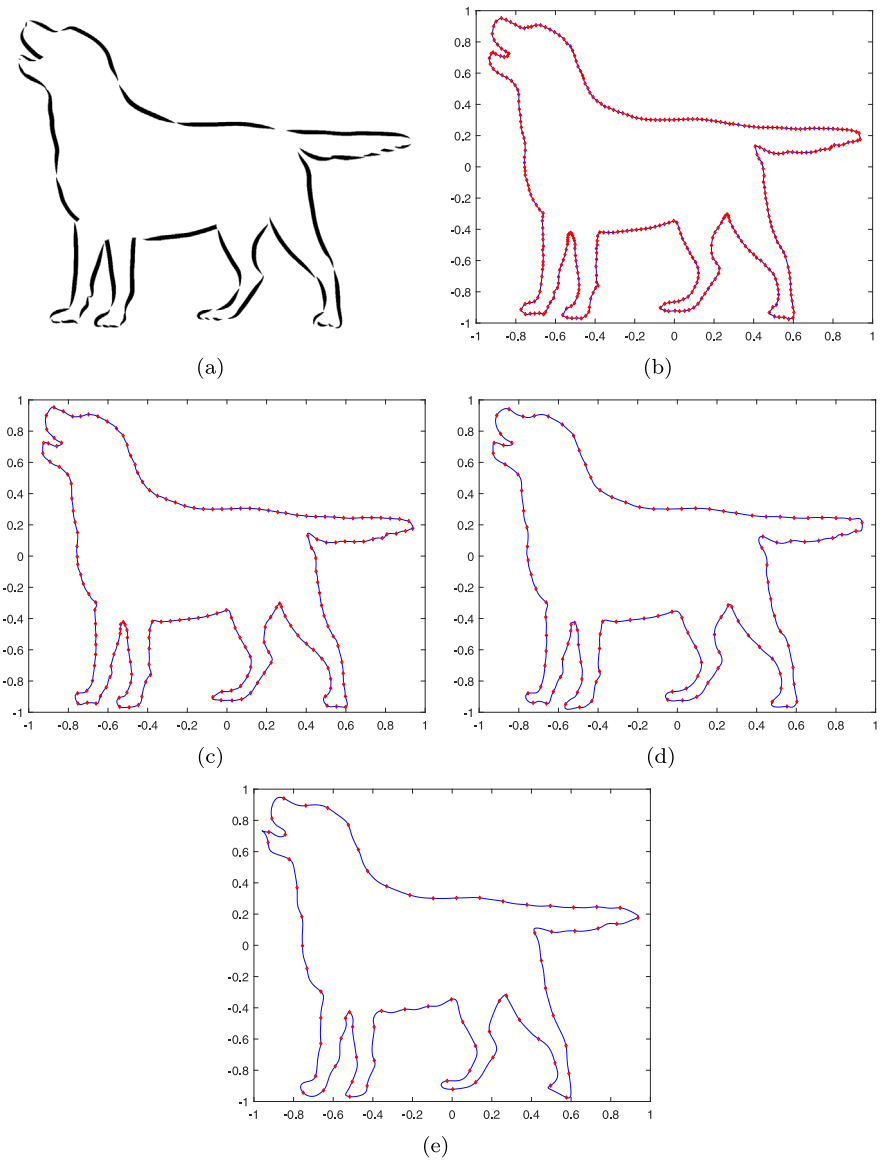


Fig. 7. Reconstruction of the sketch (a), by using 362 nodes (b), 181 nodes (c), 121 nodes (d) and 73 nodes (e).

Data availability

No data was used for the research described in the article.

Acknowledgements

The author thanks the referees for their careful reading of the manuscript and their numerous suggestions, which have improved this work. This research has been accomplished within Rete ITaliana di Approssimazione (RITA), the thematic group on "Teoria dell'Approssimazione e Applicazioni" (TAA) of the Italian Mathematical Union and partially supported by GNCS-INδAM "Finanziamenti Giovani Ricercatori 2022", (Young Researchers Fund 2022) for the project "An iterative approach for Hermite interpolants", INdAM - GNCS Project cod. CUP_E55F22000270001.

References

[1] C. Schneider, W. Werner, Hermite interpolation: The barycentric approach, Computing 46 (1) (1991) 35–51.
[2] K. Jing, Y. Liu, N. Kang, G. Zhu, A convergent family of linear Hermite barycentric rational interpolants, J. Math. Res. Appl. 40 (6) (2020) 628–646.

[3] E. Cirillo, K. Hormann, An iterative approach to barycentric rational Hermite interpolation, Numer. Math. 140 (4) (2018) 939–962.
[4] F. Dell'Accio, F. Di Tommaso, O. Nouisser, N. Siar, Rational Hermite interpolation on six-tuples and scattered data, Appl. Math. Comput. 386 (2020) 125452, 11.
[5] R. Krasny, L. Wang, A treecode based on barycentric Hermite interpolation for electrostatic particle interactions, Comput. Math. Biophys. 7 (2019) 73–84.
[6] F. Oumellal, A. Lamnii, Curve and surface construction using Hermite trigonometric interpolant, Math. Comput. Appl. 26 (1) (2021).
[7] L. Dăuş, M. Jianu, Full Hermite interpolation of the reliability of a hammock network, Appl. Anal. Discrete Math. 14 (1) (2020) 198–220.
[8] E. Cirillo, K. Hormann, On the Lebesgue constant of barycentric rational Hermite interpolants at equidistant nodes, J. Comput. Appl. Math. 349 (2019) 292–301.
[9] E. Cirillo, K. Hormann, J. Sidon, Convergence rates of a Hermite generalization of Floater-Hormann interpolants, J. Comput. Appl. Math. 371 (2020) 112624, 9.
[10] P. Henrici, Barycentric formulas for interpolating trigonometric polynomials and their conjugates, Numer. Math. 33 (2) (1979) 225–234.
[11] J.-P. Berrut, Rational functions for guaranteed and experimentally well-conditioned global interpolation, Comput. Math. Appl. 15 (1) (1988) 1–16.
[12] C. Bandiziol, S. De Marchi, On the Lebesgue constant of the trigonometric Floater-Hormann rational interpolant at equally spaced nodes, Dolomites Res. Notes Approx. 12 (2019) 51–67.
[13] P.J. Baddoo, The AAAtrig algorithm for rational approximation of periodic functions, SIAM J. Sci. Comput. 43 (5) (2021) A3372–A3392.
[14] Y. Nakatsukasa, O. Sète, L.N. Trefethen, The AAA algorithm for rational approximation, SIAM J. Sci. Comput. 40 (3) (2018) A1494–A1522.
[15] J.-P. Berrut, G. Elefante, Bounding the Lebesgue constant for a barycentric rational trigonometric interpolant at periodic well-spaced nodes, J. Comput. Appl. Math. 398 (2021) 113664, 11.
[16] P.J. Baddoo, D.G. Crowdy, Periodic Schwarz-Christoffel mappings with multiple boundaries per period, Proc. A. 475 (2228) (2019) 20190225, 20.
[17] J.-P. Berrut, G. Elefante, A periodic map for linear barycentric rational trigonometric interpolation, Appl. Math. Comput. 371 (2020) 124924, 8.
[18] R. Baltensperger, Some results on linear rational trigonometric interpolation, Comput. Math. Appl. 43 (6–7) (2002) 737–746.
[19] J.-P. Berrut, G. Elefante, A linear barycentric rational interpolant on starlike domains, Electron. Trans. Numer. Anal. 55 (2022) 726–743.
[20] F. Qi, Derivatives of tangent function and tangent numbers, Appl. Math. Comput. 268 (2015) 844–858.
[21] B.D. Welfert, Generation of pseudospectral differentiation matrices. I, SIAM J. Numer. Anal. 34 (4) (1997) 1640–1657.