Barycentric rational interpolation with no poles and high rates of approximation

Michael S. Floater · Kai Hormann

Abstract It is well known that rational interpolation sometimes gives better approximations than polynomial interpolation, especially for large sequences of points, but it is difficult to control the occurrence of poles. In this paper we propose and study a family of barycentric rational interpolants that have no real poles and arbitrarily high approximation orders on any real interval, regardless of the distribution of the points. These interpolants depend linearly on the data and include a construction of Berrut as a special case.

Mathematics Subject Classification (2000) 65D05 · 41A05 · 41A20 · 41A25

1 Introduction

A simple way of approximating a function $f : [a, b] \to \mathbb{R}$ is to choose a sequence of points

$$a = x_0 < x_1 < \cdots < x_n = b,$$

and to fit to f the unique interpolating polynomial p_n of degree at most n at these points, i.e., set

$$p_n(x_i) = f(x_i), \qquad 0 \le i \le n.$$

M. S. Floater (✉)
Centre of Mathematics for Applications, Department of Informatics, University of Oslo,
PO Box 1053, Blindern, 0316 Oslo, Norway
e-mail: michaelf@ifi.uio.no

K. Hormann
Department of Informatics, Clausthal University of Technology, Julius-Albert-Str. 4,
38678 Clausthal-Zellerfeld, Germany
e-mail: kai.hormann@tu-clausthal.de

However, as is well-known p_n may not be a good approximation to f, and for large n it can exhibit wild oscillations. For the well-documented example of Runge in which $f(x) = 1/(1+x^2)$ and the points x_i are sampled uniformly from the interval $[-5, 5]$, i.e., $x_i = -5 + 10i/n$, the sequence of polynomials (p_n) diverges as $n \to \infty$. If we are free to choose the distribution of the interpolation points x_i, one remedy is to cluster them near the end-points of the interval $[a, b]$, for example using various kinds of Chebyshev points [6].

On the other hand, if the interpolation points x_i are given to us, we have to make do with them, and then we need to look for other kinds of interpolants. A very popular alternative nowadays is to use splines (piecewise polynomials) [9], which have become a standard tool for many kinds of interpolation and approximation algorithms, and for geometric modelling. However, it has been known for a long time that the use of rational functions can also lead to much better approximations than ordinary polynomials. In fact, both polynomial and rational interpolation can exhibit exponential convergence when approximating analytic functions [1,23].

In "classical" rational interpolation, one chooses some M and N such that $M + N = n$ and fits to the values $f(x_i)$ a rational function of the form p_M/q_N where p_M and q_N are polynomials of degrees at most M and N, respectively. If n is even, it is typical to set $M = N = n/2$, and some authors have reported excellent results. The main drawback, though, is that there is no control over the occurrence of poles in the interval of interpolation.

Berrut and Mittelmann [5] suggested that it might be possible to avoid poles by using rational functions of higher degree. They considered algorithms which fit rational functions whose numerator and denominator degrees can both be as high as n. This is a convenient class of rational interpolants because, as observed in [5], every such interpolant can be written in *barycentric form*

$$r(x) = \sum_{i=0}^{n} \frac{w_i}{x - x_i} f(x_i) \bigg/ \sum_{i=0}^{n} \frac{w_i}{x - x_i} \qquad (1)$$

for some real values w_i. Thus it suffices to choose the weights $w_0, \ldots, w_n$ in order to specify r, and the idea is to search for weights which give interpolants r that have no poles and preferably good approximation properties. Various aspects of this kind of interpolation are surveyed by Berrut et al. [4].

The polynomial interpolant p_n itself can be expressed in barycentric form by letting

$$w_i = \prod_{j=0, j \neq i}^{n} \frac{1}{x_i - x_j}, \qquad (2)$$

a fact first observed by Taylor [22] and Dupuy [10], and the favourable numerical aspects of this way of evaluating Lagrange interpolants are summarized by Berrut and Trefethen [6]. Thus the weights in (2) prevent poles, but for interpolation points in general position, they do not yield a good approximation. Another option, suggested by Berrut [3], is simply to take

$$w_i = (-1)^i, \qquad i = 0, 1, \ldots, n,$$

giving

$$r(x) = \sum_{i=0}^{n} \frac{(-1)^i f(x_i)}{x - x_i} \Bigg/ \sum_{i=0}^{n} \frac{(-1)^i}{x - x_i}, \tag{3}$$

which is a truly rational function. Berrut showed that this interpolant has no poles in $\mathbb{R}$. He also used it to interpolate Runge's function and his numerical experiments suggest an approximation order of $O(1/n)$ as $n \to \infty$ for various distributions of points, including evenly spaced ones.

We independently came across the interpolant (3) while working on a method for interpolating height data given over nested planar curves [15]. Without going into details, one can view the interpolant (3) as a kind of univariate analogue of the bivariate interpolant of [15]. Our numerical examples confirmed its rather low approximation rate of $1/n$, and this motivated us to seek rational interpolants with higher approximation orders.

The purpose of this paper is to report that there is in fact a whole family of barycentric rational interpolants with arbitrarily high approximation orders which includes Berrut's interpolant (3) as a special case. The construction is very simple. Choose any integer d with $0 \le d \le n$, and for each $i = 0, 1, \ldots, n - d$, let p_i denote the unique polynomial of degree at most d that interpolates f at the $d + 1$ points $x_i, \ldots, x_{i+d}$. Then let

$$r(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x) p_i(x)}{\sum_{i=0}^{n-d} \lambda_i(x)}, \tag{4}$$

where

$$\lambda_i(x) = \frac{(-1)^i}{(x - x_i) \cdots (x - x_{i+d})}. \tag{5}$$

Thus r is a blend of the polynomial interpolants $p_0, \ldots, p_{n-d}$ with $\lambda_0, \ldots, \lambda_{n-d}$ acting as the blending functions. Note that these functions λ_i only depend on the interpolation points x_i, so that the rational interpolant r depends linearly on the data $f(x_i)$. This construction gives a whole family of rational interpolants, one for each $d = 0, 1, \ldots, n$, and it turns out that none of them have any poles in $\mathbb{R}$. Furthermore, for fixed $d \ge 1$ the interpolant has approximation order $O(h^{d+1})$ as $h \to 0$, where

$$h := \max_{0 \le i \le n-1} (x_{i+1} - x_i), \tag{6}$$

as long as $f \in C^{d+2}[a, b]$, a property comparable to spline interpolation of (odd) degree d and smoothness C^{d-1} [9]. The interpolant r can also be expressed in the barycentric form (1) and is easy and fast to evaluate in that form.

The concept of blending local approximations to form a global one is certainly not a new idea in computational mathematics. For example, Catmull and Rom [7] suggested blending polynomial interpolants using B-splines as the blending functions

(see also [2]). Shepard's method and its variants [11–13, 19, 21] for interpolating multivariate scattered data can also be viewed as blends of local interpolants, where the blending functions are based on Euclidean distance to the interpolation points. Moving least squares methods [17, 18] have become quite popular recently, where again a global approximation is formed from local ones. However, we have not seen the idea of blending developed in the context of rational interpolation and we have not seen the construction (4) in the literature. Unlike many blending methods, the blending functions λ_i in (5) do not have local support. This could be seen as a disadvantage, but on the other hand, an advantage of the interpolant r is that it is infinitely smooth.

In the following sections, we derive the main properties of the interpolant and finish with some numerical examples. As well as offering an alternative way of interpolating univariate data, we hope that these interpolants might also lead to generalizations of the bivariate interpolants of [15].

2 Absence of poles

An important property of the interpolants in (4) is that they are free of poles. In order to establish this, it will help to rewrite r as a quotient of polynomials. Multiplying the numerator and denominator in (4) by the product

$$(-1)^{n-d}(x - x_0) \cdots (x - x_n)$$

(the factor $(-1)^{n-d}$ simplifies subsequent expressions) gives

$$r(x) = \frac{\sum_{i=0}^{n-d} \mu_i(x) p_i(x)}{\sum_{i=0}^{n-d} \mu_i(x)}, \tag{7}$$

where

$$\mu_i(x) = (-1)^{n-d}(x - x_0) \cdots (x - x_n)\lambda_i(x), \tag{8}$$

or

$$\mu_i(x) = \prod_{j=0}^{i-1}(x - x_j) \prod_{k=i+d+1}^{n}(x_k - x). \tag{9}$$

Here, we understand an empty product to have value 1. Equation (7) shows that the degrees of the numerator and denominator of r are at most n and $n - d$, respectively. Since neither degree is greater than n, r can be put in barycentric form. We will treat this later in Sect. 4. Using the form of r in (7) we now show that it is free of poles. For $d = 0$ this was shown by Berrut [3].

Theorem 1 *For all d, $0 \leq d \leq n$, the rational function r in (7) has no poles in $\mathbb{R}$.*

Proof We will show that the denominator of r in (7),

$$s(x) = \sum_{i=0}^{n-d} \mu_i(x),$$ (10)

is positive for all $x \in \mathbb{R}$. Here and later in the paper it helps to define the index set

$$I := \{0, 1, \ldots, n - d\}.$$

We first consider the case that $x = x_\alpha$ for some α, $0 \leq \alpha \leq n$, and we set

$$J_\alpha := \{i \in I : \alpha - d \leq i \leq \alpha\}.$$ (11)

Then it follows from (9) that $\mu_i(x_\alpha) > 0$ for all $i \in J_\alpha$ and $\mu_i(x_\alpha) = 0$ for $i \in I \setminus J_\alpha$. Hence, since J_α is non-empty,

$$s(x_\alpha) = \sum_{i \in I} \mu_i(x_\alpha) = \sum_{i \in J_\alpha} \mu_i(x_\alpha) > 0.$$

Next suppose that $x \in (x_\alpha, x_{\alpha+1})$ for some α, $0 \leq \alpha \leq n - 1$. Then let

$$I_1 := \{i \in I : i \leq \alpha - d\}, \quad I_2 := \{i \in I : \alpha - d + 1 \leq i \leq \alpha\},$$
$$I_3 := \{i \in I : \alpha + 1 \leq i\}.$$ (12)

We then split the sum $s(x)$ into three parts,

$$s(x) = s_1(x) + s_2(x) + s_3(x), \quad \text{with} \quad s_k(x) = \sum_{i \in I_k} \mu_i(x).$$ (13)

For each $k = 1, 2, 3$, we will show that $s_k(x) > 0$ if I_k is non-empty. Since by definition $s_k(x) = 0$ if I_k is empty, and since at least one of I_1, I_2, I_3 is non-empty (since their union is I), it will then follow that $s(x) > 0$.

To this end, consider first s_2. If $d = 0$ then I_2 is empty. If $d \geq 1$ then I_2 is non-empty and from (9) we see that $\mu_i(x) > 0$ for all $i \in I_2$ and therefore $s_2(x) > 0$.

Next, consider s_3. If $\alpha \geq n - d$ then I_3 is empty. Otherwise, $\alpha \leq n - d - 1$ and I_3 is non-empty and

$$s_3(x) = \mu_{\alpha+1}(x) + \mu_{\alpha+2}(x) + \mu_{\alpha+3}(x) + \cdots + \mu_{n-d}(x).$$

Using (9) we see that $\mu_{\alpha+1}(x) > 0$, $\mu_{\alpha+2}(x) < 0$, $\mu_{\alpha+3}(x) > 0$, and so on, i.e., the first term in $s_3(x)$ is positive and after that the terms oscillate in sign. Moreover, one can further show from (9) that the terms in $s_3(x)$ decrease in absolute value, i.e.,

$$|\mu_{\alpha+1}(x)| > |\mu_{\alpha+2}(x)| > |\mu_{\alpha+3}(x)| > \cdots.$$

To see this suppose $i \geq \alpha + 1$ and compare the expression for μ_{i+1},

$$\mu_{i+1}(x) = \prod_{j=0}^{i}(x - x_j) \prod_{k=i+d+2}^{n}(x_k - x),$$

with that of μ_i in (9). Since

$$x_{i+d+1} - x > x_{i+1} - x,$$

it follows that $|\mu_i(x)| > |\mu_{i+1}(x)|$. Hence, by expressing $s_3(x)$ in the form

$$s_3(x) = \big(\mu_{\alpha+1}(x) + \mu_{\alpha+2}(x)\big) + \big(\mu_{\alpha+3}(x) + \mu_{\alpha+4}(x)\big) + \cdots,$$

it follows that $s_3(x) > 0$.

A similar argument shows that $s_1(x) > 0$ if I_1 is non-empty, for then we can express s_1 as

$$s_1(x) = \big(\mu_{\alpha-d}(x) + \mu_{\alpha-d-1}(x)\big) + \big(\mu_{\alpha-d-2}(x) + \mu_{\alpha-d-3}(x)\big) + \cdots.$$

We have now shown that $s(x) > 0$ for all $x \in [x_0, x_n]$. Finally, using similar reasoning, the positivity of s for $x < x_0$ follows from writing it as

$$s(x) = \big(\mu_0(x) + \mu_1(x)\big) + \big(\mu_2(x) + \mu_3(x)\big) + \cdots,$$

and for $x > x_n$ by writing it as

$$s(x) = \big(\mu_{n-d}(x) + \mu_{n-d-1}(x)\big) + \big(\mu_{n-d-2}(x) + \mu_{n-d-3}(x)\big) + \cdots.$$

$\square$

Having established that r has no poles, and in particular no poles at the interpolation points $x_0, \ldots, x_n$, it is now quite easy to check that r does in fact interpolate f at these points. Indeed, if $x = x_\alpha$ in (7) for some α with $0 \leq \alpha \leq n$, let J_α be as in (11). Then $p_i(x_\alpha) = f(x_\alpha)$ for all $i \in J_\alpha$, and recalling that $\mu_i(x_\alpha) > 0$ for all $i \in J_\alpha$ and $\mu_i(x_\alpha) = 0$ otherwise, and that J_α is non-empty,

$$r(x_\alpha) = \frac{\sum_{i \in J_\alpha} \mu_i(x_\alpha) p_i(x_\alpha)}{\sum_{i \in J_\alpha} \mu_i(x_\alpha)} = f(x_\alpha)\frac{\sum_{i \in J_\alpha} \mu_i(x_\alpha)}{\sum_{i \in J_\alpha} \mu_i(x_\alpha)} = f(x_\alpha).$$

We also note that r reproduces polynomials of degree at most d. For if f is such a polynomial then $p_i = f$ for all $i = 0, 1, \ldots, n - d$, and so

$$r(x) = f(x)\frac{\sum_{i=0}^{n-d} \mu_i(x)}{\sum_{i=0}^{n-d} \mu_i(x)} = f(x).$$

However, r does not reproduce rational functions. Runge's function $f(x) = 1/(1+x^2)$ is rational but its interpolant is clearly different, as can be seen from the numerical tests in Sect. 5.

3 Approximation error

Next we deal with the approximation power of the rational interpolants. Here we treat the two distinct cases $d = 0$ and $d \geq 1$ separately. The advantage in the case $d \geq 1$ is that the index set I_2 in (12) is non-empty and then we can use the partial sum $s_2(x)$ from (13) to get an error bound. Let $\|f\| := \max_{a \leq x \leq b} |f(x)|$.

Theorem 2 *Suppose $d \geq 1$ and $f \in C^{d+2}[a, b]$, and let h be as in (6). If $n - d$ is odd then*

$$\|r - f\| \leq h^{d+1}(b - a)\frac{\|f^{(d+2)}\|}{d + 2}.$$

If $n - d$ is even then

$$\|r - f\| \leq h^{d+1}\left((b - a)\frac{\|f^{(d+2)}\|}{d + 2} + \frac{\|f^{(d+1)}\|}{d + 1}\right).$$

Proof Since the error $f(x) - r(x)$ is zero whenever x is an interpolation point, it is enough to treat $x \in [a, b] \setminus \{x_0, \ldots, x_n\}$. For such x, the function $\lambda_i(x)$ in (5) is well-defined and we can express the error as

$$f(x) - r(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x)(f(x) - p_i(x))}{\sum_{i=0}^{n-d} \lambda_i(x)}.$$

Using the Newton error formula [16, Chap. 6],

$$f(x) - p_i(x) = (x - x_i) \cdots (x - x_{i+d}) f[x_i, \ldots, x_{i+d}, x],$$

where $f[x_i, \ldots, x_{i+d}, x]$ denotes the divided difference of f at the points $x_i, \ldots, x_{i+d}, x$, we thus arrive at

$$f(x) - r(x) = \frac{\sum_{i=0}^{n-d}(-1)^i f[x_i, \ldots, x_{i+d}, x]}{\sum_{i=0}^{n-d} \lambda_i(x)}. \tag{14}$$

We will derive an upper bound on the numerator and a lower bound on the denominator of this quotient. Consider first the numerator,

$$\sum_{i=0}^{n-d}(-1)^i f[x_i, \ldots, x_{i+d}, x].$$

This is a sum of $n-d+1$ terms and to avoid a bound which depends on n and therefore also h, we exploit the oscillating signs and go to divided differences of higher order. By combining the first and second terms, and the third and fourth and so on, we can express the sum as

$$-\sum_{i=0,\ i\ \text{even}}^{n-d-1}(x_{i+d+1}-x_i)f[x_i,\ldots,x_{i+d+1},x]$$

if $n-d$ is odd and

$$-\sum_{i=0,\ i\ \text{even}}^{n-d-2}(x_{i+d+1}-x_i)f[x_i,\ldots,x_{i+d+1},x]+f[x_{n-d},\ldots,x_n,x]$$

if $n-d$ is even. Then, because

$$\sum_{i=0}^{n-d-1}(x_{i+d+1}-x_i) = \sum_{i=0}^{n-d-1}\sum_{k=i}^{i+d}(x_{k+1}-x_k) \le (d+1)\sum_{k=0}^{n-1}(x_{k+1}-x_k)$$
$$= (d+1)(b-a),$$

it follows that

$$\left|\sum_{i=0}^{n-d}(-1)^i f[x_i,\ldots,x_{i+d},x]\right| \le (d+1)(b-a)\frac{\|f^{(d+2)}\|}{(d+2)!} \tag{15}$$

if $n-d$ is odd and

$$\left|\sum_{i=0}^{n-d}(-1)^i f[x_i,\ldots,x_{i+d},x]\right| \le (d+1)(b-a)\frac{\|f^{(d+2)}\|}{(d+2)!} + \frac{\|f^{(d+1)}\|}{(d+1)!} \tag{16}$$

if $n-d$ is even.

Next we consider the denominator in (14) and suppose that $x \in (x_\alpha, x_{\alpha+1})$ for some α with $0 \le \alpha \le n-1$. Because $d \ge 1$, the set I_2 in (13) is non-empty, so let j be any member of I_2. Then

$$s(x) \ge s_2(x) \ge \mu_j(x) > 0,$$

and so, by the definition of μ_i in (8),

$$\left|\sum_{i=0}^{n-d}\lambda_i(x)\right| = \frac{s(x)}{\prod_{i=0}^n|x-x_i|} \ge \frac{\mu_j(x)}{\prod_{i=0}^n|x-x_i|} = |\lambda_j(x)| = \frac{1}{|x-x_j|\cdots|x-x_{j+d}|}.$$

Since $x_j \leq x_\alpha < x < x_{\alpha+1} \leq x_{j+d}$, one has

$$|x - x_j| \cdots |x - x_{j+d}| \leq \prod_{i=j}^{\alpha} (x_{\alpha+1} - x_i) \prod_{i=\alpha+1}^{j+d} (x_i - x_\alpha)$$
$$\leq (\alpha - j + 1)! \, (d - \alpha + j)! \, h^{d+1}$$
$$\leq d! \, h^{d+1},$$

hence

$$\left| \sum_{i=0}^{n-d} \lambda_i(x) \right| \geq \frac{1}{d! \, h^{d+1}}.$$

The result now follows from this estimate combined with (15) and (16).   □

Thus for $d \geq 1$, r converges to f at the rate of $O(h^{d+1})$ as $h \to 0$, independently of how the points are distributed, as long as f is smooth enough.

In the remaining case $d = 0$ we establish a convergence rate of $O(h)$ but only under the condition that the local mesh ratio

$$\beta := \max_{1 \leq i \leq n-2} \min \left\{ \frac{x_{i+1} - x_i}{x_i - x_{i-1}}, \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}} \right\}$$

remains bounded as $h \to 0$. This agrees with what we have observed in our numerical tests: for $d = 0$ the interpolant behaves rather unpredictably when pairs of points are close together relative to the others. However, when the points are evenly spaced, β reduces to 1, and we get the unconditional convergence order $O(h)$ (or $O(1/n)$) that Berrut conjectured in [3].

Theorem 3 *Suppose $d = 0$ and $f \in C^2[a, b]$. If n is odd then*

$$\|r - f\| \leq h(1 + \beta)(b - a) \frac{\|f''\|}{2}.$$

If n is even then

$$\|r - f\| \leq h(1 + \beta)\left((b - a)\frac{\|f''\|}{2} + \|f'\|\right).$$

Proof We again employ the error formula (14). The estimates for the numerator remain valid for $d = 0$ and reduce to

$$\left| \sum_{i=0}^{n} (-1)^i f[x_i, x] \right| \leq (b - a) \frac{\|f''\|}{2}$$

if n is odd and

$$\left| \sum_{i=0}^{n} (-1)^i f[x_i, x] \right| \leq (b-a) \frac{\|f''\|}{2} + \|f'\|$$

if n is even. Thus it remains to show that the denominator in (14) satisfies the lower bound

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq \frac{1}{h(1+\beta)}. \tag{17}$$

To this end, suppose $x \in (x_\alpha, x_{\alpha+1})$ for some α with $0 \leq \alpha \leq n-1$. Since $d=0$, the partial sum $s_2(x)$ in (13) is zero and we turn to $s_1(x)$ and $s_3(x)$. Suppose first that $\alpha = n-1$. Then

$$s(x) \geq s_3(x) = \mu_n(x),$$

and so

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq |\lambda_n(x)| = \frac{1}{x_n - x} \geq \frac{1}{h},$$

which proves (17). Similarly, if $\alpha = 0$, we have

$$s(x) \geq s_1(x) = \mu_0(x),$$

and so

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq |\lambda_0(x)| = \frac{1}{x - x_0} \geq \frac{1}{h},$$

which again proves (17). Otherwise, $1 \leq \alpha \leq n-2$ and we get a bound both from s_1 and s_3. Using s_3, we have

$$s(x) \geq s_3(x) \geq \mu_{\alpha+1}(x) + \mu_{\alpha+2}(x),$$

and then

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq |\lambda_{\alpha+1}(x) + \lambda_{\alpha+2}(x)| = \frac{1}{x_{\alpha+1} - x} - \frac{1}{x_{\alpha+2} - x}$$

$$= \frac{x_{\alpha+2} - x_{\alpha+1}}{(x_{\alpha+1} - x)(x_{\alpha+2} - x)},$$

implying

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq \frac{x_{\alpha+2} - x_{\alpha+1}}{h(x_{\alpha+2} - x_\alpha)} = \frac{1}{h(1 + (x_{\alpha+1} - x_\alpha)/(x_{\alpha+2} - x_{\alpha+1}))}.$$

On the other hand, using s_1 we have

$$s(x) \geq s_1(x) \geq \mu_\alpha(x) + \mu_{\alpha-1}(x),$$

and a similar argument to the above yields

$$\left| \sum_{i=0}^{n} \lambda_i(x) \right| \geq \frac{1}{h(1 + (x_{\alpha+1} - x_\alpha)/(x_\alpha - x_{\alpha-1}))}.$$

Taking the maximum of these two lower bounds gives (17). □

4 The barycentric form

Since the degrees of the numerator and denominator of r in (7) are both at most n, we know from [5] that r can be put in the barycentric form (1). To derive this, we first write the polynomial p_i in (4) in the Lagrange form

$$p_i(x) = \sum_{k=i}^{i+d} \prod_{j=i, j \neq k}^{i+d} \frac{x - x_j}{x_k - x_j} f(x_k).$$

Substituting this into the numerator of (4) gives

$$\sum_{i=0}^{n-d} \lambda_i(x) p_i(x) = \sum_{i=0}^{n-d} (-1)^i \sum_{k=i}^{i+d} \frac{1}{x - x_k} \prod_{j=i, j \neq k}^{i+d} \frac{1}{x_k - x_j} f(x_k)$$

$$= \sum_{k=0}^{n} \frac{w_k}{x - x_k} f(x_k),$$

where

$$w_k = \sum_{i \in J_k} (-1)^i \prod_{j=i, j \neq k}^{i+d} \frac{1}{x_k - x_j}, \tag{18}$$

with J_k as in (11). This is already the form we want for the numerator of r. Similarly, for the denominator, the fact that

$$1 = \sum_{k=i}^{i+d} \prod_{j=i, j \neq k}^{i+d} \frac{x - x_j}{x_k - x_j},$$

leads to

$$\sum_{i=0}^{n-d} \lambda_i(x) = \sum_{k=0}^{n} \frac{w_k}{x - x_k}.$$

This shows that r has the barycentric form (1) with the weights $w_0, \ldots, w_n$ given by (18). This form provides an extremely simple and fast method of evaluating r. Moreover, this form can be used to evaluate derivatives of r using the derivative formulas of Schneider and Werner [20]. Since we know by Theorem 1 that r has no poles in $\mathbb{R}$, another result of Schneider and Werner [20] shows that the weights w_k must oscillate in sign. This we can now verify by observing that w_k can be written as

$$w_k = (-1)^{k-d} \sum_{i \in J_k} \prod_{j=i, j \neq k}^{i+d} \frac{1}{|x_k - x_j|}.$$

Now we look at some examples. The case $d = 1$ gives

$$w_k = (-1)^{k-1} \left(\frac{1}{x_k - x_{k-1}} + \frac{1}{x_{k+1} - x_k} \right)$$

for $1 \leq k \leq n - 1$ and

$$w_0 = \frac{-1}{x_1 - x_0}, \qquad w_n = \frac{(-1)^{n-1}}{x_n - x_{n-1}}.$$

For general d, when the points x_i are uniformly spaced with spacing h, we get

$$w_k = \frac{(-1)^{k-d}}{h^d} \sum_{i \in J_k} \frac{1}{(k-i)! \, (i+d-k)!}.$$

Since a uniform scaling of these weights does not change the interpolant r, we can multiply them by $d! \, h^d$ to give integer weights

$$w_k = (-1)^{k-d} \sum_{i \in J_k} \binom{d}{k-i}.$$

By further writing $\delta_k = (-1)^{k-d} w_k = |w_k|$, the first few sets of values $\delta_0, \ldots, \delta_n$ are

$$
\begin{array}{ll}
1, 1, \ldots, 1, 1, & d = 0, \\
1, 2, 2, \ldots, 2, 2, 1, & d = 1, \\
1, 3, 4, 4, \ldots, 4, 4, 3, 1, & d = 2, \\
1, 4, 7, 8, 8, \ldots, 8, 8, 7, 4, 1, & d = 3, \\
1, 5, 11, 15, 16, 16, \ldots, 16, 16, 15, 11, 5, 1, & d = 4.
\end{array}
$$

Thus in the uniform case, most of the weights have the same absolute value; the only change occurs near the ends of the sequence. Yet as we have shown, this "small" change increases the approximation order of the method. A similar concept is known in numerical quadrature in the form of "end-point corrections" for the composite trapezoidal rule [8, Sects. 2.8–2.9]. Note that the weights for the uniform case with $d = 1$ have also been advocated in [3] as an improvement of the case $d = 0$.

5 Numerical examples

We have tested the rational interpolants using the Matlab code for barycentric interpolation proposed by Berrut and Trefethen in [6, Sect. 7]. The basic approach to evaluating r at a given x is to check whether x is close to some x_k, within machine precision. If it is then the routine returns $f(x_k)$. Otherwise the quotient expression for $r(x)$ in (1) with (18) is evaluated. This method seems to be perfectly stable in practice. We also note that Higham [14] has shown that if the Lebesgue constant is small, Lagrange polynomial interpolation using the barycentric formula is forward stable in the sense that small errors in the data values $f(x_k)$ lead to a small relative error in the interpolant. In view of the good approximation properties of the rational interpolants r, it seems likely that they too are stable in the same sense, but this has yet to be verified.

We applied the method first to Runge's example $f(x) = 1/(1+x^2)$ for $x \in [-5, 5]$, which we sampled at the uniformly spaced points $x_i = -5+10i/n$, for various choices of n. Figure 1 shows plots of the rational interpolant with $d = 3$ for respectively,
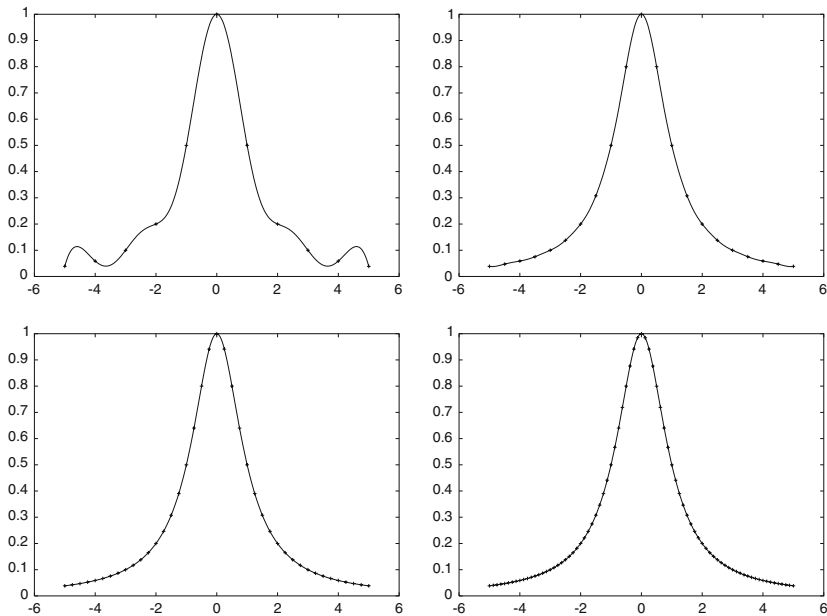

Fig. 1 Interpolating Runge's example with $d = 3$ and $n = 10, 20, 40, 80$

$n = 10, 20, 40, 80$. The second column of Table 1 shows the numerically computed errors in this example, for n up to 640, and the third column the estimated approximation orders, and they support the fourth order approximation predicted by Theorem 2. Figure 2 shows plots of the rational interpolant of the function $f(x) = \sin(x)$ at the same equally spaced points as in the previous example, but this time with $d = 4$. The fourth and fifth columns of Table 1 show the computed errors and orders, which support the fifth order approximation predicted by Theorem 2.

We also tested the method on the function $f(x) = |x|$ which has a discontinuous first derivative at $x = 0$. Figure 3 shows the rational interpolant with $d = 3$ for respectively $n = 10, 20, 40, 80$ evenly spaced points in $[-5, 5]$. The computed errors and orders of approximation can be found in the sixth and seventh columns of Table 1.

Table 1 Error in rational interpolant

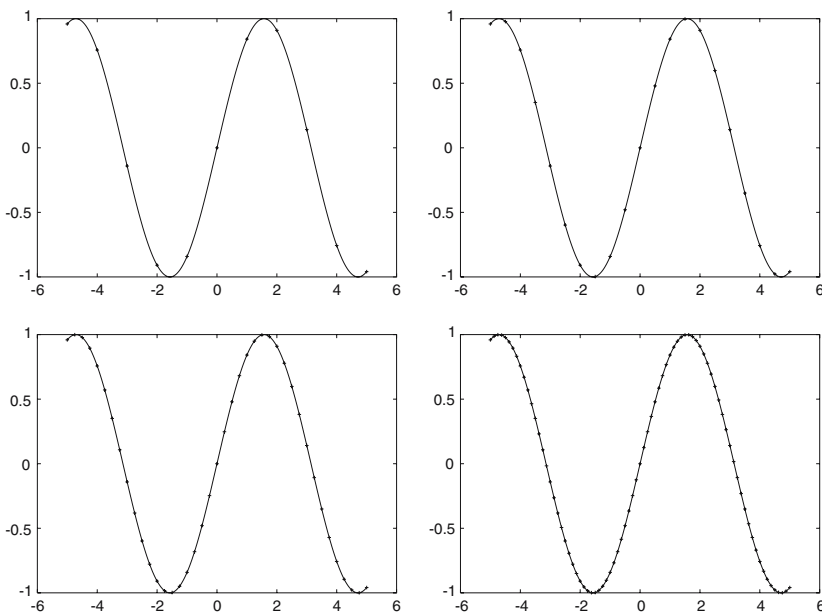| n	Runge ($d = 3$)	order	sine ($d = 4$)	order	abs ($d = 3$)	order
10	6.9e−02		1.7e−02		1.9e−01	
20	2.8e−03	4.6	3.9e−04	5.5	9.5e−02	1.0
40	4.3e−06	9.4	7.1e−06	5.8	4.8e−02	1.0
80	5.1e−08	6.4	1.3e−07	5.7	2.4e−02	1.0
160	3.0e−09	4.1	2.7e−09	5.6	1.2e−02	1.0
320	1.8e−10	4.0	6.0e−11	5.5	5.9e−03	1.0
640	1.1e−11	4.0	1.5e−12	5.3	3.0e−03	1.0

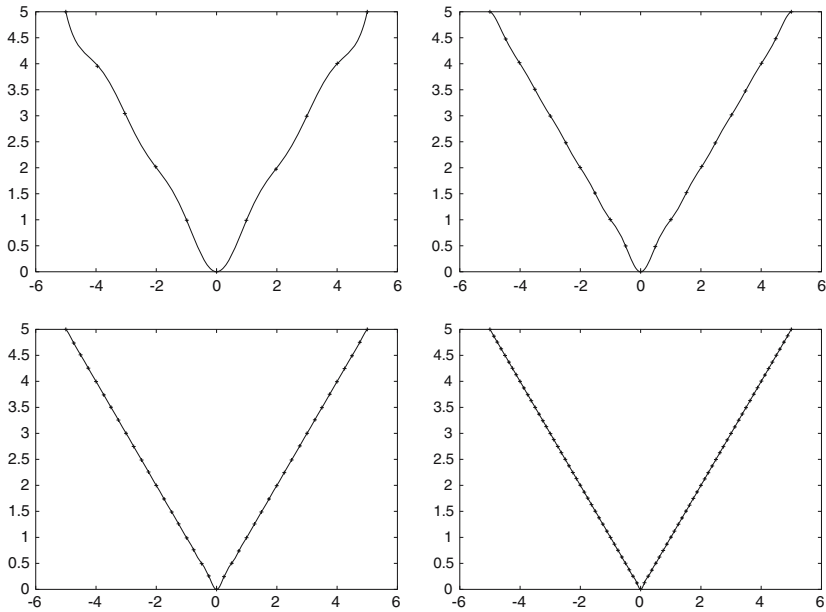**Fig. 2** Interpolating the sine function with $d = 4$ and $n = 10, 20, 40, 80$

Fig. 3 Interpolating $|x|$ over $[-5, 5]$ with $d = 3$ and $n = 10, 20, 40, 80$

Table 2 Error in Runge's example, varying d

n	best d value	error
10	$d = 0$	3.6e−02
20	$d = 1$	1.5e−03
40	$d = 3$	4.3e−06
80	$d = 7$	2.0e−10
160	$d = 10$	1.3e−15

We found that for any fixed d, the interpolants converge numerically at the rate of $O(h)$ as $h = 1/n \to 0$, which indicates that Theorem 2 really depends on f being smooth enough.

One advantage of the rational interpolants is the ease with which we can change the degree d of the blended polynomials. We can exploit this by finding the value of d which minimizes the numerically computed approximation error for a given set of points. Table 2 shows the errors in the Runge example, where, for each n, the optimal d was used. As the table shows, for this function, it is beneficial to increase d as n increases. When interpolating the sine function at the same equally spaced points it was found that $d = n$ gives the smallest error.

Finally, we make a comparison with C^2 cubic spline interpolation using clamped end conditions (i.e., taking the first derivative of the spline at the end-points equal to the corresponding derivative of the given function f). The error is $O(h^4)$ for $f \in C^4[a, b]$ (see [9, Chap. V]), the same order as for the rational interpolant with $d = 3$ (provided $f \in C^5[a, b]$). Table 3 shows the errors in the Runge example, of the two methods.

Table 3 Error in rational and spline interpolation of Runge's function

n	rational ($d = 3$)	cubic spline
10	6.9e−02	2.2e−02
20	2.8e−03	3.2e−03
40	4.3e−06	2.8e−04
80	5.1e−08	1.6e−05
160	3.0e−09	9.5e−07
320	1.8e−10	5.9e−08
640	1.1e−11	3.7e−09

Table 4 Error in rational and spline interpolation of the sine function

n	rational ($d = 3$)	cubic spline
10	1.3e−02	3.3e−03
20	1.2e−03	1.7e−04
40	8.4e−05	1.0e−05
80	5.4e−06	6.4e−07
160	3.4e−07	4.0e−08
320	2.1e−08	2.5e−09
640	1.3e−09	1.6e−10

For large n, the error in the rational interpolant is smaller than that of the spline interpolant, by a factor of more than 100, for this data set. On the other hand, when the two methods are applied to the sine function, the error in the spline interpolant is about 10 times smaller than that of the rational interpolant, as indicated in Table 4.

References

1. Baltensperger, R., Berrut, J.-P., Noël, B.: Exponential convergence of a linear rational interpolant between transformed Chebychev points. Math.Comput. **68**(227), 1109–1120 (1999)
2. Barry, P.J., Goldman, R.N.: A recursive evaluation algorithm for a class of Catmull-Rom splines. ACM SIGGRAPH Comput. Graph. **22**(4), 199–204 (1988)
3. Berrut, J.-P.: Rational functions for guaranteed and experimentally well-conditioned global interpolation. Comput. Math. Appl. **15**(1), 1–16 (1988)
4. Berrut, J.-P., Baltensperger, R., Mittelmann, H.D.: Recent developments in barycentric rational interpolation. In: de Bruin, M.G., Mache, D.H., Szabados, J., (eds) Trends and Applications in Constructive Approximation. International Series of Numerical Mathematics, vol. 151, pp 27–51. Birkhäuser, Basel (2005)
5. Berrut, J.-P., Mittelmann, H.D.: Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval. Comput. Math. Appl. **33**(6), 77–86 (1997)
6. Berrut, J.-P., Trefethen, L.N.: Barycentric Lagrange interpolation. SIAM Rev. **46**(3), 501–517 (2004)
7. Catmull, E., Rom, R.: A class of local interpolating splines. In: Barnhill R.E., Riesenfeld R.F. (eds) Computer Aided Geometric Design, pp. 317–326. Academic, New York (1974)

8. Davis, P.J., Rabinowitz, P.: Methods of numerical integration, 2nd edn. Computer Science and Applied Mathematics. Academic, Orlando (1984)

9. de Boor, C.: A practical guide to splines, revised edn. Applied Mathematical Sciences, vol. 27. Springer, Heidelberg (2001)

10. Dupuy, M.: Le calcul numérique des fonctions par l'interpolation barycentrique. Comptes Rendus de l'Académie Des Sciences. Série I, Mathématique **226**, 158–159 (1948)

11. Franke, R.: Scattered data interpolation: test of some methods. Math. Comput. **38**(157), 181–200 (1982)

12. Franke, R., Nielson, G.: Smooth interpolation of large sets of scattered data. Int. J. Numer. Methods Eng. **15**(11), 1691–1704 (1980)

13. Gordon, W.J., Wixom, J.A.: Shepard's method of "metric interpolation" to bivariate and multivariate interpolation. Math. Comput. **32**(141), 253–264 (1978)

14. Higham, N.J.: The numerical stability of barycentric Lagrange interpolation. IMA J. Numer. Anal. **24**(4), 547–556 (2004)

15. Hormann, K., Floater, M.S.: Mean value coordinates for arbitrary planar polygons. ACM Trans. Graph. **25**(4), 1424–1441 (2006)

16. Isaacson, E., Keller, H.B.: Analysis of numerical methods. Dover, New York (1994)

17. Lancaster, P., Salkauskas, K.: Surfaces generated by moving least squares methods. Math. Comput. **37**(155), 141–158 (1981)

18. Levin, D.: The approximation power of moving least-squares. Math. Comput. **67**(224), 1517–1531 (1998)

19. Ruprecht, D., Müller, H.: Free form deformation with scattered data interpolation methods. In: Farin G., Hagen H., Noltemeier H., Knödel W. (eds.) Geometric Modelling. Computing Supplementum, vol. 8, pp. 267–281. Springer, Heidelberg (1993)

20. Schneider, C., Werner, W.: Some new aspects of rational interpolation. Math. Comput. **47**(175), 285–299 (1986)

21. Shepard, D.: A two-dimensional interpolation function for irregularly-spaced data. Proceedings of the 23rd ACM national conference, pp. 517–524. ACM Press, (1968)

22. Taylor, W.J.: Method of Lagrangian curvilinear interpolation. J. Res. Nat. Bureau Stand. **35**, 151–155 (1945)

23. Tee, T.W., Trefethen, L.N.: A rational spectral collocation method with adaptively transformed Chebyshev grid points. SIAM J. Sci. Comput. **28**(5), 1798–1811 (2006)