Recent Developments in Barycentric Rational Interpolation — **Source link** ⬈

Jean-Paul Berrut, Richard Baltensperger, Hans D. Mittelmann

Institutions: University of Fribourg, Arizona State University

Topics: Interpolation, Polynomial interpolation and Barycentric coordinate system

Related papers:

- Barycentric Lagrange Interpolation

- Some new aspects of rational interpolation

- Barycentric rational interpolation with no poles and high rates of approximation

- Exponential convergence of a linear rational interpolant between transformed Chebyshev points

- Rational functions for guaranteed and experimentally well-conditioned global interpolation

Recent developments in barycentric rational interpolation

Jean–Paul Berrut, Richard Baltensperger and Hans D. Mittelmann

Abstract

In 1945, W. Taylor discovered the barycentric formula for evaluating the interpolating polynomial. In 1984, W. Werner has given first consequences of the fact that the formula usually is a rational interpolant. We review some recent advances in the use of the formula for rational interpolation.

Key words: interpolation, rational interpolation, optimal interpolation.

AMS 2000 classification: 65D05, 65D15, 41A20.

1 Introduction: polynomial interpolation in barycentric form

Interpolation is one of the central tools of (numerical) mathematics: analysis problems are being solved by first choosing from the infinite complexity of an arbitrary function a few of its (possibly unknown) values and replacing it with an interpolant of these values. The solution of the original problem is then approximated with the solution of the corresponding problem for the interpolant. Examples abound: differentiation, integration, multistep methods for ordinary differential equations, collocation methods for partial differential equations and other functional equations, etc. Interpolation is therefore a core subject of any course in numerical analysis and most books devote a chapter to it. They usually start with Lagrange interpolation, before enumerating its supposed practical drawbacks [Ber-Tre] and going over to presumably better methods.

This view of Lagrange interpolation contrasts with its ubiquity in practice and research: almost every volume of a numerical analysis journal contains some application of Lagrange cardinal functions. The present paper reviews some of the

recent advances in the *practical* use of interpolation in the form of the so–called *barycentric formula.*

We will focus on the simplest case, namely interpolation in one dimension between distinct points in an interval I. Many extensions, e.g., interpolation using information on derivatives at some or all interpolation points [Bul-Rut, Hen1] as well as interpolation in several dimensions [Bul-Rut, Bal-Ber2, She] can be handled in the barycentric setting as well.

Let thus $N+1$ distinct interpolation points (nodes) x_j, $j = 0(1)N$, be given, together with corresponding numbers f_j, $j = 0(1)N$, which can be values of a function f $\left(f_j := f(x_j)\right)$ or not. Our aim is to find an infinitely differentiable interpolant of f between the x_j. We will denote by $\mathcal{P}_m$ the linear space of all polynomials of degree at most m. The simplest problem is that of finding $p \equiv p_N \in \mathcal{P}_N$ for which

$$p(x_j) = f_j, \qquad j = 0(1)N$$

and its solution is given by the *Lagrange interpolation formula*

$$p(x) = \sum_{j=0}^{N} f_j \ell_j(x), \qquad \ell_j(x) := \frac{\prod\limits_{k \neq j}(x - x_k)}{\prod\limits_{k \neq j}(x_j - x_k)}. \tag{1.1}$$

As mentioned already, many books list drawbacks of this representation of the (unique) interpolating polynomial, in particular the fact that each evaluation of p at some x requires $\mathcal{O}(N^2)$ floating point operations (flops). However, if one has first computed the denominators of the Lagrange fundamental polynomials ℓ_j as

$$w_j := \frac{1}{\prod\limits_{k \neq j}(x_j - x_k)}, \qquad j = 0(1)N, \tag{1.2}$$

in $\mathcal{O}(N^2)$ flops, then every evaluation of p written as

$$p(x) = \ell(x) \sum_{j=0}^{N} \frac{w_j}{x - x_j} f_j, \tag{1.3}$$

where

$$\ell(x) := (x - x_0)(x - x_1) \dots (x - x_N),$$

needs only $\mathcal{O}(N)$ flops [Hen1, Ber-Tre]. ℓ is often denoted by ω or Ω in the literature. w_j is called the *weight* corresponding to the point x_j. For particular sets of points, such as equidistant or Chebyshev ones, the weights can be computed analytically [Hen1]. Nodes like Chebyshev's, distributed like $1/\sqrt{1 - x^2}$, are especially important since they lead to exponential convergence for holomorphic functions (see [Ber-Tre] for citations). Exponential convergence is a special case of *spectral convergence*, i.e., faster than polynomial.

N. Higham [Hig] has just shown that (1.3) is the formula of choice if one wishes to evaluate p as stably as possible (even when p is a very bad approximant of f). 20 years ago already, W. Werner [WeW] has given an algorithm for updating the weights w_j in $\mathcal{O}(N)$ operations when a point is added. Werner's algorithm contains an unstable expression for the weight corresponding to the new point. In [Ber-Tre], the first author* and N. Trefethen have suggested to use (1.2) for this point also, that way maintaining the stability of the formula. Their algorithm used with formula (1.3) sweeps up the last arguments against scalar Lagrange interpolation: (1.3) is *the* formula to use if one wishes the result to be as close as possible to the value of the interpolating polynomial.

Higham has also proven the stability of the so–called *barycentric formula* [Hen1, Ber-Tre]

$$p(x) = \frac{\sum_{j=0}^{N} \frac{w_j}{x - x_j} f_j}{\sum_{j=0}^{N} \frac{w_j}{x - x_j}} \tag{1.4}$$

when the set of nodes is such that interpolating constant functions is as well-conditioned a problem as the original one of interpolating f. (1.4) is obtained from (1.3) by dividing by the corresponding interpolant for the function 1 and simplifying $\ell(x)$. Since interpolating with nodes leading to ill-conditioned interpolation of constant functions is seldom indicated (see however [Gau] for an example), the barycentric formula is usually the method of choice for evaluating p.

(1.4) indeed has several advantages over (1.3) [Ber-Tre], some of which we will encounter on our way. First of all, the weights now arise in the denominator as well as in the numerator, so that any common factor independent of j may be ignored, leading to so-called *simplified weights* w_j^* [Hen1]. Most important in practice (see [Mul-Hua-Slo, Tre, Bat-Tre] and below) are the so-called *Chebyshev points of the second kind*

$$x_j = \cos j \frac{\pi}{N}, \qquad j = 0(1)N, \tag{1.5}$$

for which one simply has [Sal]

$$w_j^* = w_j^{(2)} := (-1)^j \delta_j, \qquad \delta_j = \begin{cases} 1/2, & j = 0 \text{ or } j = N, \\ 1, & \text{otherwise}, \end{cases} \tag{1.6}$$

---

*When authors are not specified we mean "of the present work".

so that the barycentric formula reads

$$p(x) = \frac{\sum_{j=0}^{N}{}'' \dfrac{(-1)^j}{x - x_j} f_j}{\sum_{j=0}^{N}{}'' \dfrac{(-1)^j}{x - x_j}}, \tag{1.7}$$

where the double prime means that the first and last terms of the sum should be halved. These factors $1/2$ at the extremities of the interval come from the fact that Chebyshev interpolation is in fact even trigonometric interpolation between equidistant points, and that on the circle the interior points arise twice, the extremal points a single time [Ber1]. Polynomials interpolating between Chebyshev points may also be evaluated rapidly and stably by means of the fast Fourier transform, albeit less transparently and more expensively for a small number of evaluation points x. Since Chebyshev nodes are located in the interval $[-1, 1]$, the interpolation problem is often considered to have been moved there, something we will also assume in most of what follows.

Formula (1.7) is extremely stable: it has been used with N up to tens of thousands [Ber2, Bat-Tre], whereas MATLAB's `polyfit` does typically not even handle $N = 50$ with the same points.

Another sign of the importance of Chebyshev points is the fact that they are clustered at the extremities of the interval of interpolation in just the right way for the corresponding linear projection to have a small norm — see [Ber-Tre] for a short description of such norms, called Lebesgue constants, and books on interpolation [Dav, Sza-Vér, Phi] for a more thorough treatment. Chapter 6 of [Mas-Han] contains further results on interpolation between Chebyshev nodes.

In this context, we draw the reader's attention to the new MATLAB software for functions, called `@chebfun` and due to Battles and Trefethen [Bat-Tre], in which an object is not a vector of the standard MATLAB, but a vector of values of a function f at enough Chebyshev points of the second kind for f to be approximated with machine precision by the corresponding interpolating polynomial. The subroutines then perform the classical operations of calculus such as differentiation, integration, etc., by Chebyshev methods.

2   From polynomial to rational interpolation

What about the barycentric formula (1.4) with points for which interpolating constant functions is ill-conditioned ? The formula will likely be unstable, the computed $\widetilde{p}$ far away from p. Is that bad ? Not necessarily! Indeed, the condition of p itself will likely be poor and p should not be used for solving problems. This is the case when the weights vary enormously, i.e., when the quotient $\frac{\max |w_j|}{\min |w_j|}$ is large [Ber-Tre]. In particular, the computed denominator (multiplied back by ℓ)

will not be the function 1, but another polynomial in $\mathcal{P}_N$. The right hand side of (1.4) becomes a rational function $r \in \mathcal{R}_{N,N}$, where $\mathcal{R}_{m,n}$ denotes the set of all rationals with numerator in $\mathcal{P}_m$ and denominator in $\mathcal{P}_n$. By the following lemma, *r interpolates the value f_j at x_j whenever $w_j \neq 0$.*

Lemma 2.1 *Let $\{(x_j, f_j)\}$, $j = 0(1)N$, be $N+1$ pairs of real numbers with $x_j \neq x_k$, $j \neq k$, and let $\{u_j\}$ be $N+1$ real numbers. Then*

a) if $u_k \neq 0$, the rational function

$$r(x) = \frac{\sum_{j=0}^{N} \frac{u_j}{x - x_j} f_j}{\sum_{j=0}^{N} \frac{u_j}{x - x_j}} \in \mathcal{R}_{N,N} \qquad (2.1)$$

interpolates f_k at x_k: $\lim_{x \to x_k} r(x) = f_k$;

b) conversely, every rational interpolant $r \in \mathcal{R}_{N,N}$ of the f_j may be written as in (2.1) for some u_j.

Proof. Statement a) is just an immediate calculation. For b), let q denote the denominator of r and let $q_j := q(x_j)$, $j = 0(1)N$. Lagrange formula (1.3) then permits to write q as $q(x) = \ell(x) \sum_{j=0}^{N} \frac{w_j q_j}{x - x_j}$. Moreover, the q_j determine the numerator as the polynomial $p \in \mathcal{P}_N$ with $p(x_j) = q_j f(x_j)$. (2.1) with $u_j := w_j q_j$, all j, then is a representation of r. ∎

a) expresses a second advantage of the barycentric formula: interpolation is warranted even when the w_j are computed with errors. This opens a wide field of research: whereas polynomial interpolation merely permits the choice of the nodes x_j, rational interpolation allows for that of the u_j also. The proof further shows that the rational interpolation problem is completely solved once the values of q at the nodes are known.

Why use rational interpolation? For two reasons, at least:

- when one cannot choose the nodes, polynomial interpolation may diverge even for well-behaved functions. An especially important case is when the f_j are the result of sampling at equidistant points. Then the sequence of p for increasing N will not converge if the f_j are the values of a function with singularities not too far from the interval of interpolation, as in Runge's example [Epp].

- even when one can choose nodes with a good distribution on the interval, such as those of Chebyshev and Legendre, polynomial interpolation

may not be the answer for it may converge much too slowly for practical purposes. Indeed, Markov's inequality implies that a polynomial of moderate degree cannot have a large slope on an interval where it does not take large values. On the interval $[-1, 1]$ the inequality says that any polynomial q of degree at most n satisfies

$$\|q'\|_\infty \leq n^2 \|q\|_\infty,$$

where $\| \cdot \|_\infty$ denotes the maximum or L_∞–norm. It follows that, for a polynomial p of degree n to be a good approximation of f with p' a good approximation of f', one must have

$$n \geq \sqrt{\|f'\|_\infty / \|f\|_\infty}.$$

Approximating a function that behaves like $\mathrm{erf}(\delta x)$, where erf denotes the error function, therefore requires too large an N with very large δ (see Figure 3 in [Ber-Mit2]).

The classical answer to these difficulties is to use piecewise functions as approximants. Our interest here, however, is to stay with analytic functions. They are simpler to program and often converge faster, i.e., exponentially or spectrally [Ber-Tre]. Rational interpolation is the next such possibility, and the function $1/(1 + ax^2)$ for large a shows that no limitation such as Markov's inequality is in the way there.

3   Classical rational interpolation

The problem here is to find $r \in \mathcal{R}_{m,n}$ that interpolates the f_j, i.e., $p \in \mathcal{P}_m$ and $q \in \mathcal{P}_n$ such that

$$r(x_j) = \frac{p(x_j)}{q(x_j)} = f_j, \qquad j = 0(1)N. \tag{3.1}$$

In the canonical representation, p and q together have $m + n + 2$ coefficients, of which one may be set to 1 by dividing both polynomials by it. The $N + 1$ interpolation conditions (3.1) thus are equally numerous as the coefficients when

$$N = m + n. \tag{3.2}$$

This condition characterizes the classical rational interpolation problem, see [Sto] (or [Gut] for a more general treatment). Grosse's catalogue [Gro] contains a long list of papers on rational interpolation.

The problem need not have a solution (see the examples in [Sto, p. 50] or [Ber-Mit1, p. 367]), but it usually does and, if so, then the solution is unique [Sto, p. 51] and may be written in barycentric form (Lemma 2.1).

3.1   Classical rational interpolation in barycentric form

Schneider and Werner [Sch-Wer] have been the first to determine barycentric representations of rational interpolants. Their method uses a classical way of determining the Newton form of the interpolant before applying an algorithm of Werner to pass from the Newton to the barycentric form. In [Ber-Mit1], the first and last authors have given a method for directly finding the corresponding weights u_j when $n \leq m$.

Theorem 3.1 *If a solution r of the classical rational interpolation problem (3.1)–(3.2) with $n \leq m$ exists, then $\mathbf{u} = [u_0, u_1, \ldots, u_N]$ is a vector of weights in one of its barycentric representations (2.1) iff $\mathbf{u}$ belongs to the kernel of the $N \times (N+1)-$matrix*

$$
\mathbf{A} := \begin{bmatrix}
1 & 1 & 1 & \cdots & 1 \\
x_0 & x_1 & x_2 & \cdots & x_N \\
x_0^2 & x_1^2 & x_2^2 & \cdots & x_N^2 \\
\vdots & \vdots & \vdots & & \vdots \\
x_0^{m-1} & x_1^{m-1} & x_2^{m-1} & \cdots & x_N^{m-1} \\
f_0 & f_1 & f_2 & \cdots & f_N \\
f_0 x_0 & f_1 x_1 & f_2 x_2 & \cdots & f_N x_N \\
f_0 x_0^2 & f_1 x_1^2 & f_2 x_2^2 & \cdots & f_N x_N^2 \\
\vdots & \vdots & \vdots & & \vdots \\
f_0 x_0^{n-1} & f_1 x_1^{n-1} & f_2 x_2^{n-1} & \cdots & f_N x_N^{n-1}
\end{bmatrix} . \tag{3.3}
$$

In order to save space, we introduce some notation: $\mathbf{V}_{P,Q}$, $P \leq Q$, will be the matrix made up of the first P rows of the transposed Vandermonde matrix corresponding to the $Q + 1$ values $x_0, \ldots, x_Q$, and $\mathbf{F}_Q = \mathrm{diag}(f_0, \ldots, f_Q) \in \mathbb{R}^{Q+1,Q+1}$ will be the diagonal matrix of values $f_0, \ldots, f_Q$. Then $\mathbf{A}$ in (3.3) may be written $\mathbf{A} = [\mathbf{V}_{m,N}^T, \mathbf{F}_N \mathbf{V}_{n,N}^T]^T$.

The proof of Theorem 3.1 consists in showing that the first m equations express that the degree of the denominator written in the form $\ell(x) \sum_{j=0}^{N} \frac{u_j}{x - x_j}$ in (2.1) is at most n, the last n equations that the numerator degree is at most m. Lemma 2.1 a) implies that r in (2.1) interpolates in x_j if $u_j \neq 0$; the latter is not necessary, though.

As a corollary, Theorem 3.1 delivers the kernel of the matrix $\mathbf{V}_{N-1,N}$: it is just the space spanned by the vector of the polynomial barycentric weights w_j.

The customary way of coping with the case $n > m$ is to determine the reciprocal of r by interpolating the values $1/f_j$. This requires a special treatment in case some of the f_j vanish. Brimmeyer [Bri] has discovered how to stay within the barycentric context by considering the corresponding x_j as poles and using the method given in [Ber5] and described below to preassign such poles in the determination of the weights of $1/r$.

In our experience, the algorithm given in [Ber-Mit1] for computing the kernel of $\mathbf{A}$ in (3.3) is much more efficient than computing the singular value decom-

position of $\mathbf{A}$ with MATLAB's routine `svd`. It consists in triangulating (3.3) in two steps: one analytical, which leads to divided differences, the other numerical, through Gaussian elimination with column pivoting. Though, in contrast with p, r is a good approximation even with equidistant interpolation points for N large enough, Chebyshev nodes again lead to much better conditioned problems. Note, however, that these nodes must be reordered for a stable computation of the divided differences. The algorithm is then extremely stable, see the examples in [Ber-Mit1].

As with the interpolating polynomial, the degrees of p and q may be smaller than m and n. This manifests itself in the kernel of $\mathbf{A}$ having dimension larger than 1. A way of coping with this, suggested in [Ber-Mit1], is to decrease n by 1, restart the computation, and repeat until a kernel of dimension 1 is obtained for some $n^* < n$. $\mathbf{u}$ then yields the barycentric representation (2.1) of a *reduced* r, i.e., one in which the linear factors corresponding to common zeros of p and q have been simplified. We will denote it by r^*. If $u_k = 0$ for some k and $r^*(x_k) \neq f_k$, then x_k is an *unattainable point* [Sto] and the problem does not have a solution. This is the first part of the following theorem, due to Schneider and Werner [Sch-Wer, Ber-Mit1], which allows an easy detection of two drawbacks of classical rational interpolation directly from the weights.

Theorem 3.2 *Let u be barycentric weights of a reduced rational interpolant $r^* = p^*/q^*$. Then*

a) *a point x_k is unattainable iff $u_k = 0$;*

b) *if $u_k \neq 0$ for all k, if the interpolation points have been ordered as $x_0 < x_1 < \ldots < x_N$ and if $\operatorname{sign} u_j = \operatorname{sign} u_{j+1}$, then r^* has an odd number of poles between x_j and x_{j+1}.*

Proof. a) follows from the above discussion and Lemma 2.1a, b) from noticing that q^* changes sign, thus has a zero, between x_j and x_{j+1}, and p^* does not at the same place, for otherwise r^* would not be reduced. ∎

$u_k = 0$ in (2.1) simply means that the node x_k is ignored. One may then eliminate the pair (x_k, f_k) and solve the problem in $\mathcal{R}_{m,n}$, $m+n = N-1$, $n = n^*-1$ [Ber5]. (Such elimination of data might be problematic if several u_j vanish.)

Zhu and Zhu [Zhu-Zhu] have recently suggested an equivalent, and maybe even more elegant way of finding r by directly determining the values $\mathbf{q} := [q_0, q_1, \ldots, q_N]^T$ of its denominator. Let $\mathbf{W} := \operatorname{diag}(w_0, w_1, \ldots, w_N)$ be the diagonal matrix of the polynomial weights (1.2) to $x_0, x_1, \ldots, x_N$. Then the proof of Lemma 2.1 shows that one has the bijective map $\mathbf{u} = \mathbf{W}\mathbf{q}$; the kernel equation $\mathbf{A}\mathbf{u} = \mathbf{0}$ may thus be written $\mathbf{A}\mathbf{W}\mathbf{q} = \mathbf{0}$.

Corollary 3.3 [Zhu-Zhu] *If a solution r of the classical rational interpolation problem (3.1)–(3.2) with $n \leq m$ exists, then the vector $\mathbf{q}$ of its denominator values belongs to the kernel of the $N \times (N+1)$–matrix $\mathbf{A}\mathbf{W}$.*

The elements of $\mathbf{A}\mathbf{W}$ are given in [Zhu-Zhu].

3.2 Reduced complexity barycentric rational interpolation

Besides its many advantages, the barycentric representation of $r \in \mathcal{R}_{m,n}$ also has some disadvantages in comparison with the canonical $r(x) = (a_m x^m + \ldots + a_1 x + a_0)/(b_n x^n + \ldots + b_1 x + b_0)$. One of them is the fact that its evaluation at a particular point x requires about twice as many flops.

In [Ber6], the first author has suggested a method for improving upon this. Since the numerator and the denominator have degrees at most $L := \max\{m, n\}$, they may both be written as interpolating polynomials between any number of points larger than L. Let therefore M with $L \leq M \leq N$ be given. By Lemma 2.1b, the solution of problem (3.1) may be written as

$$r(x) = \sum_{j=0}^{M} \frac{u_j}{z - z_j} f_j \bigg/ \sum_{j=0}^{M} \frac{u_j}{z - z_j}. \tag{3.4}$$

(3.4) has been called a *reduced complexity barycentric representation of* r in [Ber6]. With it, only $M + 1$ unknown u_j, $j = 0(1)M$, are to be determined, but interpolation is guaranteed by Lemma 2.1b only at the corresponding x_j, called *primary nodes* in [Ber6]. The degree conditions that make up the matrix $\mathbf{A}$ in Theorem 3.1 are also less numerous, since the denominator, respectively numerator degree must be decreased merely by $M - n$, resp. $M - m$ units. In total, $N - M$ less weights are to be determined, but $2(N - M)$ less degree diminishing conditions hold. The remaining $N - M$ equations are the interpolation conditions at the *secondary nodes* $x_{M+1}, x_{M+2} \ldots, x_N$. Interpolation is warranted at any of these x_k by [Ber6]

$$\sum_{j=0}^{M} f[x_j, x_k] u_j = 0, \tag{3.5}$$

where, as usual, $f[x_{i_0}, \ldots, x_{i_s}]$ denotes the divided difference of order s for the points $x_{i_0}, \ldots, x_{i_s}$.

The matrix $\mathbf{B}$ with elements $b_{ij} := f[x_i, x_j]$, $i = 0(1)M$, $j = M + 1(1)N$, of the divided differences in (3.5) is the *Löwner matrix* corresponding to the sets of points $\{x_0, \ldots, x_M\}$ and $\{x_{M+1}, \ldots, x_N\}$.

Theorem 3.4 *If a solution r of the classical rational interpolation problem (3.1)– (3.2) with $n \leq m$ exists, and if $x_0, \ldots, x_M$ are the primary nodes, then any vector $\mathbf{u} = [u_0, \ldots, u_M]$ of its weights in a reduced complexity barycentric representation (3.4) belongs to the kernel of the $M \times (M + 1)$–matrix*

$$\mathbf{A} = \begin{bmatrix} \mathbf{V}_{M-n,M} \\ \mathbf{V}_{M-m,M} \mathbf{F}_M \\ \mathbf{B} \end{bmatrix},$$

where $\mathbf{B}$ is the Löwner matrix given above.

The partitioning of the nodes in primary and secondary ones is important. Numerical experiments demonstrate in particular that the extreme points should be primary nodes, for otherwise the interpolation at the secondary ones, enforced by the equations $\mathbf{Bu} = \mathbf{0}$, may not be accurate.

The decisive advantages of the reduced complexity representation are the smaller number of points in the (often ill–conditioned) Vandermonde matrices and the fact that the divided differences arising in the computation of the kernel of $\mathbf{A}$ have only $M - n + 2$ arguments, as opposed to $N - n + 1$ when working with the representation (2.1) and the matrix (3.3). The improvement in equidistant interpolation may be spectacular, see Table 3 in [Ber6].

Steffen [Ste] has adapted the above reduced complexity rational interpolation to the method of Zhu and Zhu, thereby computing only the $M + 1$ values of the denominator at the primary nodes. Her numerical results are very close to those of [Ber6].

3.3   Monitoring the poles

The poles are at the same time the curse and the blessing of classical rational interpolation. Their bright side, mentioned above, is their capability of accomodating large gradients of f. Their dark side is the fact that, for N small, they may show up about everywhere in the plane, thus also in the immediate vicinity of, or even on, the interval of interpolation when interpolating a perfectly innocuous function (see Cordellier's example in [WeH] or [Ber-Mit2]; one may also read p. 357 of [Ber-Mit1]). When more is known about f than just the interpolated points in the plane, one should therefore try to incorporate the extra information into the interpolant to monitor the poles.

Sometimes the location of some or all of the poles is known a priori (see §8 below for an example). The above method for determining the barycentric weights may be modified to determine the rational interpolant in this case [Ber5]. Assume that P poles of the denominator are prescribed; denote them by z_k, $k = 1, \ldots, P$ and their multiplicity by ν_k with $\nu := \sum_{k=1}^{P} \nu_k$; assume further that $z_k \neq x_j$ for all j and all k. The problem is now to compute rational interpolants with prescribed poles, i.e., to find

$$r = p/q \in \mathcal{R}_{m,n+\nu}, \qquad m + n = N, \qquad n + \nu \leq N,$$

such that (3.1) is satisfied and r has the ν preassigned poles. If the interpolant exists, its denominator will contain as a factor the polynomial

$$d(z) := a \prod_{k=1}^{P} (z - z_k), \quad a \neq 0 \in \mathbb{C} \text{ arbitrary.} \tag{3.6}$$

Let

$$d_j := d(x_j), \qquad j = 0(1)N, \tag{3.7}$$

be the values of d at the nodes. Then the part of r remaining to be determined,

$$r^* = p^*/q^* := r \cdot d \in \mathcal{R}_{m,n}, \qquad m \geq n,$$

must take the values

$$r^*(x_j) = g_j, \qquad g_j := f_j \cdot d_j, \qquad j = 0(1)N, \tag{3.8}$$

at the nodes. Accounting for the difficulties of unattainable points and/or multiple solutions mentioned above, we are left with the following problem:

(R) *Find the largest possible $n^* \leq n$ and the corresponding unique $r^* \in \mathcal{R}_{m^*,n^*}$ with $m^* + n^* = N$, $n^* \leq m^*$, that satisfies the interpolation conditions (3.8).*

Theorem 3.1 implies as a corollary that, if such a r^* exists, its barycentric weights $\mathbf{b} = [b_0, b_1, \ldots, b_N]^T$ make up the one–dimensional kernel of the matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{V}_{m^*,N} \\ \mathbf{V}_{n,N}\mathbf{G}_N \end{bmatrix},$$

where $\mathbf{G}_N = \mathrm{diag}(g_0, g_1, \ldots, g_N)$. Once this kernel has been determined, e.g., by the algorithm of [Ber-Mit1], barycentric weights of r in (3.1) are given by [Ber5]

$$u_j = d_j \cdot b_j,$$

a result extended in the following theorem proven in [Ber5].

Theorem 3.5 *If some $r \in \mathcal{R}_{m,n+\nu}$, $n \leq m \leq N$, $n+\nu \leq N$, exist with $r(x_j) = f_j$, $j = 0(1)N$, and with poles of order ν_k at the points z_k, $k = 1(1)P$, then one of them is given by (3.1) with $u_j = d_j b_j$. On the other hand, if*

$$\sum_{j=0}^{N} \prod_{k \neq j} (x_j - z_k) f_j \neq 0, \tag{3.9}$$

r as in (2.1) with $u_j = d_j b_j$ has a pole at z_k.

The proof implies that, if the numerator has degree $\leq N - \nu$, the conditions (3.9) are not satisfied, meaning that r cannot be guaranteed to display the pre-scribed poles. This may reflect the fact that there is no $r \in \mathcal{R}_{N,N}$ interpolating the values f_j and possessing those poles. For instance, for $c \in \mathbb{R}$ constant there is no $r \not\equiv c$ interpolating the values $f_j \equiv c \; \forall \; j$, since every r as in (3.1) with constant f_j is this same constant for all choices of u_j. This is reasonable from an approximation point of view. Whether (3.9) is necessary for the presence of a pole at z_k is an open question.

When all the poles are prescribed, one has $n = 0$, r^* is the interpolating polynomial and one gets the following corollary, which will be important in the subsequent sections.

Corollary 3.6 *If $r \in \mathcal{R}_{N,\nu}$, $\nu \le N$, exists with $r(x_j) = f_j$, $j = 0(1)N$, and with poles of order ν_k at z_k, $k = 1(1)P$, $\nu := \sum_{k=1}^{P} \nu_k$, then its barycentric weights are given (up to a constant factor) by $u_j = w_j b_j$, $j = 0(1)N$.*

r is then the quotient of the polynomial of degree at most N interpolating $f \cdot d$ between the x_j and the polynomial d [Ber-Mit2]. It cannot have any other pole than the z_k, which eliminates the potentially harmful free poles of classical rational interpolation. The rational interpolants arising in the optimal approximation of functionals in the Hardy space H^2 make an interesting example [Ber3].

4   Optimal attachment of poles to the interpolating polynomial

Corollary 3.6 shows that, when one can prescribe all of the poles, rational interpolation is as simple as polynomial interpolation. Though in some cases the poles can be obtained a priori (see below), this is usually not the case. In [Ber-Mit2], the first and last authors have suggested a way of determining an optimal position of the poles *when the function f taking the values $f(x_i) = f_i$ is known everywhere on the interval I* (as in the application of §9).

Consider the (potential) poles z_k, $k = 1(1)P$, as variables in a rational interpolant

$$r(x) = \frac{\displaystyle\sum_{j=0}^{N} \frac{w_j d_j}{x - x_j} f_j}{\displaystyle\sum_{j=0}^{N} \frac{w_j d_j}{x - x_j}} = \frac{\displaystyle\sum_{j=0}^{N} \frac{w_j \prod_{k=1}^{P}\left(1 - \dfrac{x_j}{z_k}\right)}{x - x_j} f_j}{\displaystyle\sum_{j=0}^{N} \frac{w_j \prod_{k=1}^{P}\left(1 - \dfrac{x_j}{z_k}\right)}{x - x_j}} \tag{4.1}$$

with denominator d as in (3.6) (the second equality of (4.1) only holds when $z_k \ne 0 \; \forall \; k$ and is obtained by dividing by $\prod_{k=1}^{P}(-z_k)$ — to simplify the notation, we consider from here on a pole of multiplicity ν_k as ν_k separate poles). The goal is to choose the z_k in such a way that the interpolant r in (4.1) is as good an approximation of f as possible. It has been suggested in [Ber-Mit2] to consider functions continuous on I and to minimize the infinity norm of $r - f$, i.e., to solve the following min–max, or nonlinear Chebyshev, approximation problem:

(A) *Minimize $\|r - f\|_\infty := \max_{x \in I} |r(x) - f(x)|$, with r as in (4.1), with respect to the z_k, $k = 1(1)P$.*

Theorem 4.1 [Ber-Mit2] *Problem (A) always has a solution.*

As mentioned in [Ber-Mit2], the solution is usually not unique (constant functions are counterexamples). The more interesting question of the unicity of the optimal interpolant r is open.

These interpolants have very nice properties. There can be no unattainable point nor unwanted pole. r is always at least as good as the interpolating polynomial, for the latter is the case when all z_k are at infinity (see the right-hand side of (4.1)). Moreover, attachment of another pole never results in a worsening of the approximation, since the already optimized poles constitute a feasible point for the optimization.

The optimization problem has been numerically solved with success in [Ber-Mit2] with standard modern optimization algorithms. The nice properties just mentioned occur in practice, and the numerical results, e.g., with Cordellier's example, for which classical rational interpolation with small N is useless, are quite impressive.

We mention that the authors of [Ber-Mit2] originally intended to solve the more ambitious problem of minimizing $\|r-f\|_\infty$ with respect to all u_j in a representation (2.1) of r, but that they encountered difficulties, both theoretical (existence of an optimum, of an alternating sequence, etc.) and practical ones (too many parameters to optimize).

5 Differentiation of rational interpolants

Already in their 1986 paper [Sch-Wer], Schneider and Werner have given very elegant general formulae for the derivatives of rational interpolants in the form (2.1). These formulae, one for differentiation between the nodes and one at the nodes, cover arbitrary orders of differentiation. Outside the nodes they are just barycentric formulae with the values f_i at the nodes replaced by divided differences of the corresponding order. For simplicity, we give them here only for the first and second order derivatives that will be needed in an application below:

$$r'(x) = \begin{cases} \displaystyle\sum_{j=0}^{N} \frac{u_j}{x-x_j} r[x,x_j] \bigg/ \sum_{j=0}^{N} \frac{u_j}{x-x_j}, & x \neq x_i,\ i=0(1)N, \\[4mm] -\bigg(\displaystyle\sum_{\substack{j=0 \\ j\neq i}}^{N} u_j r[x_i,x_j]\bigg)\bigg/ u_i, & x = x_i \end{cases} \tag{5.1a}$$

and

$$r''(x) = \begin{cases} 2\displaystyle\sum_{j=0}^{N} \frac{u_j}{x-x_j} r[x,x,x_j] \bigg/ \sum_{j=0}^{N} \frac{u_j}{x-x_j}, & x \neq x_i,\ i=0(1)N, \\[4mm] -2\bigg(\displaystyle\sum_{\substack{j=0 \\ j\neq i}}^{N} u_j r[x_i,x_i,x_j]\bigg)\bigg/ u_i, & x = x_i, \end{cases} \tag{5.1b}$$

with $r[z,z,x_j] = \frac{r'(z)-r[z,x_j]}{z-x_j}$.

6   From nonlinear to linear rational interpolation

With all approximants given in § 3 and 4 except that for which all the poles are pre-scribed (Corollary 3.6), the barycentric weights u_j, and therefore the denominator, depend on the interpolated function: the approximation operator is nonlinear. In practice, however, many problems are addressed by means of linear approximants: also nonlinear ones are often solved with a sequence of linear approximations.

   To every fixed set of nodes x_j and every set of corresponding fixed weights u_j there corresponds a linear interpolant: for example, with $u_j = w_j \ \forall \ j$ this is just polynomial interpolation. For Chebyshev points such as (1.5), the polynomial weights (1.2) lead to a very well-conditioned polynomial interpolant, whereas this notoriously is not the case with equidistant nodes. *It seems therefore natural to try finding good weights corresponding to a given set of nodes* $\{x_j\}$.

   For a given vector of nodes, the vector of weights $\mathbf{b} = [b_0, b_1, \ldots, b_N]^T$ defines the linear vector space $\mathcal{R}_N^{(b)}$ of all rational interpolants (1.4) with these weights [Bal-Ber2]. The set of rational functions

$$L_j^{(b)}(x) := \frac{b_j}{x - x_j} \bigg/ \sum_{k=0}^{N} \frac{b_k}{x - x_k}, \quad j = 0, 1, \ldots, N, \tag{6.1}$$

constitutes a basis of $\mathcal{R}_N^{(b)}$.

   The first author has suggested in [Ber2] to use the weights (1.6), i.e., the interpolant (1.7), for every set of nodes. This linear rational interpolant never has poles in the interval of interpolation and is extremely well-conditioned in practice, even with random points. Unfortunately, its convergence is not fast enough for approximating higher order derivatives for many sets of nodes, in particular for the very important equidistant grid (the first author will give a description of this convergence in a forthcoming paper). There are, however, certain sets of nodes for which it converges about as fast as the best polynomial interpolant: conformal maps of Chebyshev nodes !

7   Conformal point shifts

Polynomial interpolation between Chebyshev points is trigonometric interpolation of even functions between equidistant points [Ber1, Tre]. It has very nice proper-ties: fast convergence for very smooth functions, small operator norm (Lebesgue constant), very stable barycentric formula. It also has some drawbacks, in partic-ular the $\mathcal{O}(N^{-2})$–concentration of the nodes at the extremities of the interval of interpolation, which results in (at least) three difficulties:

   a) ill-conditioning of the derivatives near the extremities;

   b) bad distribution of the information over the interval;

    c) mediocre approximation of functions with shocks close to the center, where points are scarcer.

    Several experts in the solution of differential equations by means of Chebyshev interpolants [Bay-Tur, Boy, Kos-Tal] have suggested to address these difficulties by a conformal shift of the nodes toward the equidistant position. This does not change the order of convergence of the interpolant.

    To be more precise, consider, beside the x–space in which f is to be approximated, another space, with variable y, and the $N+1$ Chebyshev points of the second kind $y_j = \cos j\pi/N$, $j = 0(1)N$, on the interval $J := [-1, 1]$ in this y–space. Let further g be a conformal map from a domain $\mathcal{D}_1$ containing J (in the y–space) to a domain $\mathcal{D}_2$ containing I (in the x–space). This defines new interpolation points on I, $x_j := g(y_j)$, and the conformal transplantation [Hen2] $F(y) := f(x)$ of any function in the x–space back into y–space. (The transplantation of an x–space–function will be denoted by the corresponding upper case letter.) Then, one may consider at least two approximants of a function f:

– the transplantation a_N of the polynomial A_N interpolating F between the y_j:

$$A_N(y) := \sum_{j=0}^{N} F(y_j)L_j(y) = \sum_{j=0}^{N} f(x_j)L_j\big(g^{-1}(x)\big) =: a_N(x); \qquad (7.1)$$

– the rational interpolant mentioned at the end of §6:

$$r_N(x) = \sum_{j=0}^{N} \frac{w_j^{(2)}}{x - x_j} f_j \Bigg/ \sum_{j=0}^{N} \frac{w_j^{(2)}}{x - x_j}, \qquad w_j^{(2)} \text{ from (1.6).} \qquad (7.2)$$

    The most favorable case is that in which the function f to be interpolated is analytic in a domain containing I. It follows from a classical result that, if $f : \mathcal{D}_2 \mapsto \mathbb{C}$ is such that the composition $f \circ g : \mathcal{D}_1 \mapsto \mathbb{C}$ is analytic inside and on an ellipse E_ρ with foci at ± 1 and sum of its axes equal to 2ρ, $\rho > 1$, then [For, p. 28]

$$|a_N(x) - f(x)| = |A_N(y) - F(y)| = \mathcal{O}(\rho^{-N}) \text{ for every } x \in [-1, 1].$$

The corresponding result for r_N in (7.2) has been proven in [Bal-Ber-Noë]:

Theorem 7.1 *Let $\mathcal{D}_1$, $\mathcal{D}_2$ be two domains of $\mathbb{C}$ containing $J = [-1, 1]$, respectively I ($\subset \mathbb{R}$), let g be a conformal map $\mathcal{D}_1 \to \mathcal{D}_2$ such that $g(J) = I$, and f be a function $\mathcal{D}_2 \to \mathbb{C}$ such that the composition $f \circ g : \mathcal{D}_1 \to \mathbb{C}$ is analytic inside and on an ellipse C_ρ ($\subset \mathcal{D}_1$), $\rho > 1$, with foci at ± 1 and with the sum of its major and minor axes equal to 2ρ. Let r_N be the rational function (7.2) interpolating f between the transformed Chebyshev points $x_k := g(y_k)$. Then, for every $x \in [-1, 1]$,*

$$|r_N(x) - f(x)| = \mathcal{O}(\rho^{-N}).$$

Conformal point shifts thus preserve exponential convergence. They may markedly lessen (though not eliminate) the difficulties a)–c) enumerated at the section's onset. The ill-conditioning of the derivatives near extremal Chebyshev points is due to the accumulation of points there, which for large N are so close that a small change in a f_j has a strong impact on the derivatives around x_j. (This may be quantitatively studied with the pseudospectrum, see [Tre, p. 108].) Following Kosloff and Tal-Ezer, one may improve upon this by moving the points closer to equidistant. With the approximation (7.1), the derivatives are given by

$$a'_N(x) = A'_N(y) \cdot \left[g^{-1}(x)\right]' = \frac{A'_N(y)}{g'(y)} \tag{7.3a}$$

and

$$a''_N(x) = \frac{1}{[g'(y)]^2} A''_N(y) - \frac{g''(y)}{[g'(y)]^3} A'_N(y), \tag{7.3b}$$

in which $A'_N(y)$ and $A''_N(y)$ may be computed by (5.1). With the approximation (7.2), the derivatives are simply given by the formulae (5.1) with $u_j = w_j^{(2)} \ \forall \ j$.

Kosloff and Tal-Ezer have suggested the map

$$g(y) = \frac{\arcsin(\alpha y)}{\arcsin \alpha}, \qquad 0 < \alpha < 1. \tag{7.4}$$

In the limiting cases, $\alpha \to 0$ maintains the points at their Chebyshev position, whereas $\alpha \to 1$ renders them equidistant. The derivatives of g to be used in (7.3) are given by

$$g'(y) = \frac{\alpha}{\arcsin \alpha} \frac{1}{\sqrt{1 - (\alpha y)^2}}, \qquad g''(y) = \frac{\alpha^3}{\arcsin \alpha} \frac{y}{\sqrt{\left(1 - (\alpha y)^2\right)^3}},$$

so that in (7.3b)

$$\frac{g''(y)}{[g'(y)]^3} = (\arcsin^2 \alpha) y.$$

The effect of this map upon the derivatives has been extensively studied, see, e.g., [Red-Wei-Nor, Mea-Ren, Abr-Gar]. It is indeed significant. And the map is even more successful in alleviating the drawbacks b) and c) mentioned above: through the refurbishment of the center with nodes it approximates functions with steep gradients or oscillations there much better than the simple polynomial interpolating between Chebyshev points, see the examples in [Ber-Mit4].

One may combine such changes of variable with the optimal attachment of poles of §4. In [Ber-Mit4], the first and last authors have done this by attaching poles v_k to A_N in y–space, which corresponds to attaching poles $z_k := g(v_k)$ to a_N in x–space. For a function with a steep gradient in the center of the interval, such as Hemker's example

$$f(x) = \cos \pi x + \frac{\mathrm{erf}(\delta x)}{\mathrm{erf}(\delta)}, \qquad \delta = \sqrt{250}, \qquad -1 \le x \le 1,$$

the improvement obtained by means of the poles is still more pronounced than that coming from the change of variable. However, this is no longer the case with a function oscillating in the center: the optimization problem with a number of poles big enough to cope with a large number of oscillations seems beyond the capability of existing optimization software.

The second author has just noticed that the pole attachment may usually also be performed with r_N from (7.2) by multiplying every weight $w_j^{(2)}$ by $\prod_k(x_j - z_k)$. The resulting rational interpolant is the quotient of two rational functions with the same denominator as r_N, one interpolating $f \cdot d$, the other interpolating d from (3.6). Contrary to the attachment via (7.1), further poles than the z_k may arise in $\mathbb{C}$, but they move infinitely far as $N \to \infty$. One advantage is the more elegant formula (5.1) for the derivatives which avoids the chain rule. Exponential convergence is maintained, as will be shown in further work. The numerical examples in [Bal-Ber2] suggest that this interpolant should be just a little less good (no more than one digit) than a_N with the same attached poles.

8 An application: the linear rational pseudospectral method for boundary value problems

In order to demonstrate the use of linear rational interpolation in practical problems we shall now solve the following simple boundary value problem: find u on the interval $[-1, 1]$ that satisfies the differential equation

$$u''(x) + p(x)u'(x) + q(x)u(x) = h(x) \tag{8.1a}$$

at every $x \in (-1, 1)$ and takes the values

$$u(-1) = u_\ell, \qquad u(1) = u_r \tag{8.1b}$$

at the boundary points. We assume that the functions p, q, and h are such that the problem is well-posed.

Suppose first that one knows a good location for poles z_k of an approximation of u, e.g., because the equation is fuchsian or from an application of the WKB–method [Wei, Bal-Ber-Dub]. To solve (8.1), one may then try substituting for u a linear rational interpolant $\widetilde{u}$ written in the basis (6.1),

$$\widetilde{u}(x) = \sum_{j=0}^{N} u_j L_j^{(b)}(x), \tag{8.2}$$

to obtain

$$\sum_j u_j L_j^{(b)''}(x) + p(x) \sum_j u_j L_j^{(b)'}(x) + q(x) \sum_j u_j L_j^{(b)}(x) = h(x). \tag{8.3}$$

Since the boundary values should enter the solution of the problem, one takes here a set of nodes containing -1 and 1, e.g., Chebyshev points of the second kind. Then (8.3) yields an equation for the unknown values $u_1, \ldots, u_{N-1}$ of $\widetilde{u}$ at $x_1, \ldots, x_{N-1}$ (u_0 and u_N being known from the boundary conditions). If the exact solution does not miraculously happen to belong to $\mathcal{R}_N^{(b)}$, (8.3) does not have a solution (8.2) and one may collocate, i.e., merely require that the two sides of (8.3) agree in as many values x_i of x as there are unknowns, here $N - 1$:

$$\sum_j u_j L_j^{(b)''}(x_i) + p(x_i) \sum_j u_j L_j^{(b)'}(x_i) + q(x_i) \sum_j u_j L_j^{(b)}(x_i) = h(x_i), \ i = 1(1)N-1.$$

$$(8.4)$$

This is a system of linear equations for the unknown values $u_1, \ldots, u_{N-1}$, which may be written as

$$\mathbf{A}\mathbf{u} = \mathbf{h} \tag{8.5}$$

with $\mathbf{A} := \mathbf{D}^{(2)} + \mathbf{P}\mathbf{D}^{(1)} + \mathbf{Q}$ and

$$\mathbf{u} := [u_1, u_2, \ldots, u_{N-1}]^T,$$
$$\mathbf{D}^{(1)} = \left(D_{ij}^{(1)}\right), \quad D_{ij}^{(1)} := L_j^{(b)'}(x_i),$$
$$\mathbf{D}^{(2)} = \left(D_{ij}^{(2)}\right), \quad D_{ij}^{(2)} := L_j^{(b)''}(x_i),$$
$$\mathbf{P} := \mathrm{diag}\left(p(x_i)\right), \quad \mathbf{Q} := \mathrm{diag}\left(q(x_i)\right),$$
$$\mathbf{h} := [h(x_i) - u_r\left(L_0^{(b)''}(x_i) + p(x_i)L_0^{(b)'}(x_i)\right) - u_\ell\left(L_N^{(b)''}(x_i) + p(x_i)L_N^{(b)'}(x_i)\right)]^T,$$
$$i, \ j = 1, \ldots, N - 1.$$

An advantage of the barycentric representation of $\widetilde{u}$ is the simplicity of the formulae for the elements of $\mathbf{D}^{(1)}$ and $\mathbf{D}^{(2)}$, which may be given as

$$D_{ij}^{(1)} = \begin{cases} \dfrac{b_j/b_i}{x_i - x_j}, & i \neq j, \\ -\sum\limits_{k \neq i} D_{ik}^{(1)}, & i = j, \end{cases}$$

$$D_{ij}^{(2)} = \begin{cases} 2D_{ij}^{(1)}\left(D_{ii}^{(1)} - \dfrac{1}{x_i - x_j}\right), & i \neq j, \\ -\sum\limits_{k \neq i} D_{ik}^{(2)}, & i = j. \end{cases} \tag{8.6}$$

The weights b_j enter explicitly only through the remarkably simple formula for the non-diagonal elements of $\mathbf{D}^{(1)}$! The use of these formulae was first advocated in [Bal-Ber1] for the polynomial case, in [Bal-Ber2] and [Ber-Bal] for the rational case. They may be obtained from (5.1); a simple direct proof is given in [Ber-Tre].

The matrix $\mathbf{A}$ in (8.5) is full, as opposed to those of the finite difference and the finite element methods. The reason for the efficiency of the method, already in the polynomial case, lies in the spectral convergence of $\widetilde{u}$ toward u — for good

nodes — when all functions arising in the problem are analytic within ellipses containing the interval $[-1, 1]$ in their interior, see the examples in [Tre]. This remains true when one takes $b_j = w_j d_j$ as in (4.1) to solve a problem whose solution has the same poles z_k:

Theorem 8.1 [Bal-Ber-Dub] *Let the solution u of (8.1) be meromorphic with poles at $z_1, \ldots, z_P$. Then the linear rational collocation method with trial function (8.2) for Chebyshev points of the second kind converges exponentially toward u, and at least as fast as the corresponding polynomial collocation solution of an associated boundary value problem for $u \cdot d$.*

The associated boundary value problem is explictly known [Pér-Cas-Hay, Ber-Bal-Dub]. Of particular importance is the fact that prescribing the poles results in a better solution for small N. Indeed, when using spectral elements in several dimensions, one cannot increase N at will in each element since the system of equations would become too large. The figure displays the solution of Example 2 of [Bal-Ber-Dub] with $N = 17$ together with the polynomial solution (left) and the rational solution with the correct poles (right). Note in particular that the u_j (big dots) are very good approximations to $u(x_j)$ in the second case: they could be interpolated with splines to avoid oscillations for N small. Weideman has proposed in [Wei] a method that is mathematically equivalent to ours when all the poles are fixed a priori but that does not employ the barycentric representation.

<div align="center">
Polynomial   Rational
</div>

Comparison of two pseudospectral solutions of a boundary value problem (8.1)

Every set of points x_j and every set of b_j determine another linear rational collocation method. A natural choice are shifted points x_j and the weights $b_j = w_j^{(2)} \ \forall \ j$. $\mathcal{R}_N^{(b)}$ then is the set of all interpolants (7.2) and takes advantage of the improved condition of the derivatives and/or of the better approximation in the center of I, see §7.

The better condition of the derivatives is not so important when solving (8.5) directly, that is by Gaussian elimination, for this amounts to applying $\mathbf{A}^{-1}$, i.e., to integration [Ber4, Tan-Tru]. However, it might be important when solving (8.5) for more complicated problems, and is definitely an advantage when one applies $\mathbf{A}$ itself, e.g.:

— when solving time evolution partial differential equations with the pseudospectral *method of lines* (pseudospectral discretization in space followed by a time-stepping algorithm such as extrapolation or Runge–Kutta for solving the resulting system of ordinary differential equations in time). This is the application that led Kosloff and Tal-Ezer to advocate their shift (7.4).
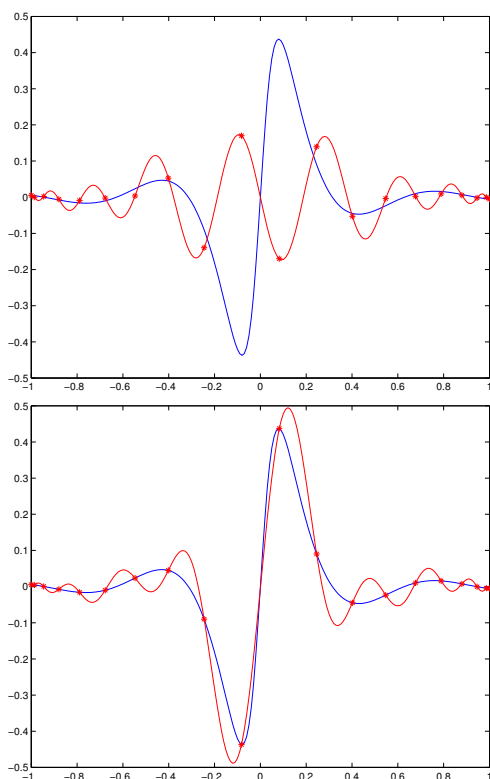
The use of the corresponding points x_j in the just mentioned $\mathcal{R}_N^{(b)}$ has been studied in [Bal-Ber2]. It decreases the number of time steps by a factor of two to three without any noticeable change in the computing effort for spectral discretization.

— when solving spatial systems of equations such as (8.5) by iteration [Ber-Bal]: a good choice of the parameter α in (7.4) (about $\alpha = .99$) may decrease the number of iterations by a third, and this again with no change in the computer code but the command computing the x_j.

9 Back to nonlinear: adaptive point shifts and poles

As already mentioned in §7, a side effect of Kosloff and Tal-Ezer's point shift is the improvement in the approximation of functions with steep gradients in the center of I by polynomials interpolating between Chebyshev points; in many examples

it becomes the most desirable effect. It is too blunt, however: it materializes only when there is only one front and when the latter lies precisely in the center. This is by no means the usual case: fronts may arise everywhere in I and there may be several of them.

To cope with the first difficulty, Bayliss and Turkel [Bay-Tur] have studied some two-parameter shifts, with one parameter for the location of the front, the other for its intensity. The following observation then led the first and last authors to a solution of the several fronts problem: with one steep front, the inverse change of variable is itself steep at the front, but has very small slope away from it, so that adding another inverse shift preserves both slopes. This naturally leads to the inverse shift [Ber-Mit5]

$$y(x) = g^{[-1]}(x) = \mu + \frac{1}{\lambda} \sum_{q=1}^{Q} \arctan[\alpha_q(x - \beta_q)], \tag{9.1}$$

where the parameters β_q and α_q determine location and intensity of the q–th front, and where λ and μ ensure that $g^{[-1]}(-1) = -1$ and $g^{[-1]}(1) = 1$. Whenever needed, the shift $g(y)$ itself is obtained by inverting $g^{[-1]}$, a simple task in the two-front case [Ber-Mit5]. In many instances, as in the solution of differential equations below, only $g^{[-1]}$ (or, rather, its derivatives) arises in the solution of the collocation problem; g is just needed for evaluating the final solution between the nodes.

For approximation, if one has some (possibly vague) information about localization and intensity of the front, one may try adjusting the parameters α_q and β_q by trial and error; this can be quite effective in practice. In [Ber-Mit5], the parameters were optimized in a more sophisticated way by minimizing

$$\|R - F\|_\infty := \max_{y \in [-1,1]} |R(y) - F(y)|,$$

where R denotes the rational approximation in y–space, with modern simulated annealing software. The effect of the shift is quite impressive for interior fronts, much more pronounced than that of the optimized poles.

The latter are still useful, though. Firstly, they may bring a noticeable bit of extra precision at the fronts. Secondly, and more importantly, they are necessary for accomodating boundary layers. Indeed, by moving the points toward the interior fronts, the shifts have the negative effect of depleting the extremities, thus worsening the approximation of incidental boundary layers. Poles, which may be located everywhere and in particular have real part outside I, remain efficient in that case, as the results of [Ber-Mit5] demonstrate. We just reproduce here Table 1, whose captions should be self explanatory and may be found in [Ber-Mit5].

β	α	Poles	$\|r - f\|$	$\|r' - f'\|$	$\|r'' - f''\|$
$*$	$*$		$1.867\mathrm{e}-1$	$2.361\mathrm{e}+1$	$3.454\mathrm{e}+3$
$*$	$*$	$(-.4902, \pm 2.011\mathrm{e}-2)$ $(-.5056, \pm 2.228\mathrm{e}-2)$ $(-.5178, \pm 5.613\mathrm{e}-2)$	$5.224\mathrm{e}-4$	$2.158\mathrm{e}-1$	$8.241\mathrm{e}+1$
$-.5185$	7.408		$9.447\mathrm{e}-9$	$5.012\mathrm{e}-6$	$1.138\mathrm{e}-2$
$-.4976$	8.273	$(-1.027, \pm 3.147\mathrm{e}-3)$	$1.279\mathrm{e}-11$	$6.270\mathrm{e}-9$	$1.654\mathrm{e}-5$
$-.4981$	8.519	$(-1.030, \pm 3.574\mathrm{e}-3)$ $(1.062, \pm 5.346\mathrm{e}-3)$	$2.495\mathrm{e}-12$	$1.754\mathrm{e}-9$	$5.659\mathrm{e}-6$

Effect of an optimized Bayliss–Turkel point shift on
rational approximation with and without optimized poles in an example with $N=100$

The advances in approximation brought by optimized poles and point shifts may be used for improving solutions of problems. As an example we take again the boundary value problem (8.1). By means of the chain rule one sees that the point shift transplants (8.1a) into the equation

$$[y'(x)]^2 U''(y) + [y''(x) + P(y)y'(x)]U'(y) + Q(y)U(y) = H(y) \qquad (9.2)$$

in y–space, which will be solved by collocation at Chebyshev points of the second kind y_j.

The first and last authors have suggested in [Ber-Mit3] and [Ber-Mit6] the following two-step recursive procedure for solving (9.2):

Step 1. Compute the approximate solution $\mathbf{U}^{(k)} = [U_1^{(k)}, \ldots, U_{N-1}^{(k)}]^T$ of (9.2) — with the boundary conditions (8.1b) — by the linear rational collocation method with $b_j = w_j d_j$, $d_j = \prod_{k=1}^{P}(y_j - v_k)$, with the poles $v_1, \ldots, v_P$ in y–space ($d_j \equiv 1$ for $k = 1$) and the inverse point shift (9.1). This amounts to solving a system (8.5) with matrix

$$\mathbf{A} := \mathbf{G}_1^2 \mathbf{D}^{(2)} + (\mathbf{G}_2 + \mathbf{G}_1 \mathbf{P})\mathbf{D}^{(1)} + \mathbf{Q},$$

where $\mathbf{D}_1$ and $\mathbf{D}_2$ are the Chebyshev differentiation matrices (8.6) and where $\mathbf{G}_1$ and $\mathbf{G}_2$ denote the diagonal matrices of the derivatives of $g^{[-1]}$ at the nodes x_i,

$$\mathbf{G}_1 = \mathrm{diag}\big(y'(x_1), \ldots, y'(x_{N-1})\big), \ \ \mathbf{G}_2 = \mathrm{diag}\big(y''(x_1), \ldots, y''(x_{N-1})\big),$$

while $\mathbf{P}$ and $\mathbf{Q}$ contain the values $P(y_i) = p(x_i)$, resp. $Q(y_i) = q(x_i)$.

Step 2. Minimize the residual norm

$$\|[y']^2 R'' + [y'' + Py']R' + QR - H\|_\infty$$

of the differential equation for the rational interpolant

$$R(y) := \sum_{j=0}^{N} \frac{w_j^{(2)} \prod_{\ell=1}^{P} \left(1 - \frac{y_j}{v_\ell}\right)}{y - y_j} U_j^{(k)} \Bigg/ \sum_{j=0}^{N} \frac{w_j^{(2)} \prod_{\ell=1}^{P} \left(1 - \frac{y_j}{v_\ell}\right)}{y - y_j}$$

with respect to the poles v_k in y–space, $k = 1, \ldots, P$, and the shift parameters α_q, β_q, $q = 1, \ldots, Q$. This changes the b_j to yield a new interpolant to the $U_j^{(k)}$.

The derivatives $y'(x)$ and $y''(x)$ required in the computation of $\mathbf{G}_1$ and $\mathbf{G}_2$ are given by the simple formulae

$$y'(x) = \frac{1}{\lambda} \sum_{q=1}^{Q} \frac{\alpha_q}{1 + s_q^2},$$

$$y''(x) = -\frac{2}{\lambda} \sum_{q=1}^{Q} \frac{\alpha_q^2 s_q}{(1 + s_q^2)^2}, \quad s_q := \alpha_q(x - \beta_q).$$

In all the examples we tried [Ber-Mit6] the method approximates the solution with an error merely about ten times as large as the direct approximation of the exact solution in [Ber-Mit5] (one example of which is given in the table above), a splendid performance. The convergence of the method has not been proven yet; however, an L_2–Galerkin version has been shown to reduce the energy norm of the error at each step of the algorithm.

10 Conclusion

We hope that the present article has convinced the reader that applications of the barycentric representation of rational interpolants brings interesting advances in infinitely smooth practical approximation. Its use in classical rational interpolation yields a very stable way of computing the interpolant and allows for a relatively simple detection of unattainable points and poles. The latter may also be easily monitored in the complex plane and their location optimized to yield new rational interpolants which approximate a given function with an error that diminishes with the number of the poles. In view of the globality of the interpolants, fronts are handled with conformal shifts of variables which may be optimized as well. Though expensive to determine, the resulting approximants display an impressive accuracy. They may be used in the solution of problems such as differential and integral equations.

References

[Abr-Gar] Abril–Raymundo M.R., García–Archilla B., Approximation properties of a mapped Chebyshev method, *Appl. Numer. Math.* **32** (2000) 119–136.

[Bal-Ber1] Baltensperger R., Berrut J.–P., The errors in calculating the pseudo-spectral differentiation matrices for Čebyšev–Gauss–Lobatto points, *Comput. Math. Appl.* **37** (1999) 41–48. Errata **38** (1999) 119.

[Bal-Ber2] Baltensperger R., Berrut J.–P., The linear rational collocation method, *J. Comput. Appl. Math.* **134** (2001) 243–258.

[Bal-Ber-Dub] Baltensperger R., Berrut J.–P., Dubey Y., The linear rational pseudospectral method with preassigned poles, *Numer. Algorithms* **33** (2003) 53–63.

[Bal-Ber-Noël] Baltensperger R., Berrut J.–P., Noël B., Exponential convergence of a linear rational interpolant between transformed Chebyshev points, *Math. Comp.* **68** (1999) 1109–1120.

[Bat-Tre] Battles Z., Trefethen L. N., An extension of MATLAB to continuous functions and operators, *SIAM J. Sci. Comput* **25** (2004) 1743–1770.

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

[Ber1] Berrut J.–P., Baryzentrische Formeln zur trigonometrischen Interpolation (I), *Z. angew. Math. Phys. (ZAMP)* **35** (1984) 91–105.

[Ber2] Berrut J.–P., Rational functions for guaranteed and experimentally well–conditioned global interpolation, *Comput. Math. Appl.* **15** (1988) 1–16.

[Ber3] Berrut J.–P., Barycentric formulae for some optimal rational approximants involving Blaschke products, *Computing* **44** (1990) 69–82.

[Ber4] Berrut J.–P., *A pseudospectral Čebyšev method with preliminary transform to the circle: ordinary differential equations*, Report No. 252, Mathematisches Institut, Technische Universität München, 1990; revised Université de Fribourg, 1995.

[Ber5] Berrut J.–P., The barycentric weights of rational interpolation with prescribed poles, *J. Comput. Appl. Math.* **86** (1997) 45–52.

[Ber6] Berrut J.–P., A matrix for determining lower complexity barycentric representations of rational interpolants, *Numer. Algorithms* **24** (2000) 17–29.

[Ber-Bal] Berrut J.–P., Baltensperger R., The linear rational pseudospectral method for boundary value problems, *BIT* **41** (2001) 868–879.

[Ber-Mit1] Berrut J.–P., Mittelmann H., Matrices for the direct determination of the barycentric weights of rational interpolation, *J. Comput. Appl. Math.* **78** (1997) 355–370.

[Ber-Mit2] Berrut J.–P., Mittelmann H., Rational interpolation through the optimal attachment of poles to the interpolating polynomial, *Numer. Algorithms* **23** (2000) 315–328.

[Ber-Mit3] Berrut J.–P., Mittelmann H. D., The linear rational pseudospectral method with iteratively optimized poles for two-point boundary value problems, *SIAM J. Sci. Comput.* **23** (2001) 961–975.

[Ber-Mit4] Berrut J.–P., Mittelmann H. D., Point shifts in rational interpolation with optimized denominator, in: J. Levesley, I.J. Anderson and J.C. Mason, eds., *Algorithms for Approximation IV, Proceedings of the 2001 International Symposium* (The University of Huddersfield, 2002) 420–427.

[Ber-Mit5] Berrut J.–P., Mittelmann H. D., Adaptive point shifts in rational approximation with optimized denominator, *J. Comput. Appl. Math.* **164–165** (2004) 81–92.

[Ber-Mit6] Berrut J.–P., Mittelmann H. D., Optimized point shifts and poles in the linear rational pseudospectral method for boundary value problems, to appear in *J. Comput. Phys.*

[Ber-Tre] Berrut J.–P., Trefethen L. N., Barycentric Lagrange interpolation, *SIAM Rev.* **46** (2004) 501–517.

[Boy] Boyd J. P., The arctan/tan and Kepler-Burgers mappings for periodic solutions with a shock, front, or internal boundary layer, *J. Comput. Phys.* **98** (1992) 181–193.

[Bri] Brimmeyer, M., *Forme barycentrique de complexité réduite pour interpolants rationnels avec pôles prescrits* (Master's thesis, University of Fribourg (Switzerland), 2001).

[Bul-Rut] Bulirsch R., Rutishauser H., Interpolation und genäherte Quadratur, in: Sauer R., Szabó I., Hsg., *Mathematische Hilfsmittel des Ingenieurs* (Grundlehren der math. Wissenschaften Bd. 141, Springer, Berlin–Heidelberg, 1968) 232–319.

[Dav] Davis P. J., *Interpolation and Approximation* (Dover, New York, 1975).

[Epp] Epperson J.F., On the Runge example, *Amer. Math. Monthly* **94** (1987) 329–341.

[For] Fornberg B., *A Practical Guide to Pseudospectral Methods* (Cambridge Univ. Press, Cambridge 1996).

[Gau] Gautschi W., Moments in quadrature problems, *Comput. Math. Appl.* **33** (1997) 105–118.

[Gro] Grosse E., A catalogue of algorithms for approximation, in: Mason J.C., Cox M.G. (eds.), *Algorithms for Approximation II* (Chapman and Hall, London, 1990) 479–514.

[Gut] Gutknecht M. H., The rational interpolation problem revisited, *Rocky Mt. J. Math.* **21** (1991) 263-280.

[Hen1] Henrici P., *Essentials of Numerical Analysis with Pocket Calculator Demonstrations* (Wiley, New York, 1982).

[Hen2] Henrici P., *Applied and Computational Complex Analysis, Vol. 3* (Wiley, New York, 1986).

[Hig] Higham N., The numerical stability of barycentric Lagrange interpolation, *IMA J. Numer. Anal.* **24** (2004) 547-556.

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

[Mas-Han] Mason, J. C., Handscomb, D. C., *Chebyshev Polynomials* (Chapman & Hall, Boca Raton, 2003).

[Mea-Ren] Mead J. L., Renaut R. A., Accuracy, resolution, and stability properties of a modified Chebyshev method, *SIAM J. Sci. Comput.* **24** (2002) 143–160.

[Mul-Hua-Slo] Mulholland L. S., Huang W. Z., Sloan D. M., Pseudospectral solution of near-singular problems using numerical coordinate transformations based on adaptivity, *SIAM J. Sci. Comput.* **19** (1998) 1261–1289.

[Pér-Cas-Hay] Pérez-Acosta F., Casasús L., Hayek N., Rational collocation for linear boundary value problems, *J. Comput. Appl. Math.* **33** (1990) 297-305.

[Phi] Phillips G. M., *Interpolation and Approximation by Polynomials* (Springer, New York, 2003).

[Red-Wei-Nor] Reddy S. C., Weideman J. A. C., Norris G. F., *On a modified Chebyshev pseudospectral method*, Report, Oregon State University, 1999.

[Sal] Salzer H. E., Lagrangian interpolation at the Chebyshev points $x_{n,\nu} = \cos(\nu\pi/n)$, $\nu = 0(1)n$; some unnoted advantages, *The Computer J.* **15** (1972) 156–159.

[Sch-Wer] Schneider C., Werner W., Some new aspects of rational interpolation, *Math. Comp.* **47** (1986) 285–299.

[She] Shepard D., A two–dimensional interpolation function for irregularly–spaced data, *Proc. 23rd Nat. Conf. ACM* (1968) 517–524.

[Ste] Steffen S., *Eine Methode zur direkten Bestimmung der Nennerwerte rationaler Interpolierender mit reduzierter Komplexität* (Master's thesis, University of Fribourg (Switzerland), 2002).

[Sto] Stoer J., *Einführung in die Numerische Mathematik I* (4. Aufl., Springer, Berlin–Heidelberg–New York, 1983).

[Sza-Vér] Szabados J., Vértesi P., *Interpolation of Functions* (World Scientific, Singapore, 1990).

[Tan-Tru] Tang T., Trummer M. R., Boundary layer resolving pseudospectral methods for singular perturbation problems, *SIAM J. Sci. Comput.* **17** (1996) 430–438.

[Tre] Trefethen L. N., *Spectral Methods in MATLAB* (SIAM, Philadelphia, 2000).

[Wei] Weideman J. C., Spectral methods based on nonclassical orthogonal polynomials, in: Gautschi W. et al., eds., *Applications and computation of orthogonal polynomials* (Birkhäuser, ISNM 131, 1999) 239-252.

[WeH] Werner H., Algorithm 51: A reliable and numerically stable program for rational interpolation of Lagrange data, *Computing* **31** (1983) 269–286.

[WeW] Werner W., Polynomial interpolation: Lagrange versus Newton, *Math. Comp.* **43** (1984) 205–217.

[Zhu-Zhu] Zhu X., Zhu G., A method for directly finding the denominator values of rational interpolants, *J. Comput. Appl. Math.* **148** (2002) 341–348.

Département de Mathématiques
Université de Fribourg
CH–1700 Fribourg/Pérolles
Switzerland
E–mail address: Jean-Paul.Berrut@unifr.ch

Département de Mathématiques
Université de Fribourg
CH–1700 Fribourg/Pérolles
Switzerland
E–mail address: Richard.Baltensperger@unifr.ch

Department of Mathematics and Statistics
Arizona State University
Tempe, Arizona 85287–1804
USA
E–mail address: Mittelmann@asu.edu