

Matrices for the direct determination of the barycentric weights of rational interpolation

Jean-Paul Berrut

Université de Fribourg, Mathématiques, CH-1700 Fribourg/Pérolles, Switzerland

and

Hans D. Mittelmann

Department of Mathematics, Arizona State University, Tempe, Arizona 85287-1804, USA

Abstract.- Let x_0, \dots, x_N be $N + 1$ values of a real (or complex) variable x . Every rational interpolant r of a function f with numerator and denominator degrees $\leq N$ can be written in its barycentric form

$$r(x) = \sum_{k=0}^N \frac{u_k}{x - x_k} f_k \bigg/ \sum_{k=0}^N \frac{u_k}{x - x_k}$$

which is completely determined by a vector \mathbf{u} of its $N + 1$ barycentric weights u_k . Finding \mathbf{u} is therefore an alternative to the determination of the coefficients in the canonical form of r ; it is advantageous inasmuch as \mathbf{u} contains information about unattainable points and poles.

In classical rational interpolation the numerator and the denominator of r are made unique (up to a constant factor) by restricting their respective degrees. We determine here the corresponding vectors \mathbf{u} by applying an elimination algorithm to a matrix whose kernel is the space of the \mathbf{u} 's.

Subject classification: AMS(MOS) Primary 65D05, 41A05; Secondary 41A20.

Key words: interpolation, rational interpolation, barycentric representation, barycentric weights.

1. The problem

Let x_0, x_1, \dots, x_N be $N + 1$ distinct points (nodes) in \mathbb{R} , f_0, f_1, \dots, f_N corresponding values in \mathbb{R} (\mathbb{C}). For any two given integers $m, n \geq 0$ we will denote by $\mathcal{R}_{m,n}$ the set of all rational functions with numerator degree $\leq m$ and denominator degree $\leq n$.

The (classical) rational interpolation problem is the following: given m and n , find

$$r = \frac{p}{q} \in \mathcal{R}_{m,n} \quad (1)$$

such that

$$r(x_k) = \frac{p(x_k)}{q(x_k)} = f_k, \quad k = 0(1)N. \quad (2)$$

It is well known that one may assume without loss of generality that $n \leq m$. Indeed, reorder the nodes in such a way that the μ of them at which $f_k = 0$ are first. If $m^* := m - \mu \leq n$, interpolate the f_k between x_μ, \dots, x_N by $r^* \in \mathcal{R}_{m^*,n}$, otherwise interpolate the $\frac{1}{f_k}$ by $r^* \in \mathcal{R}_{n,m^*}$. In the first case, $r^* \prod_{k=0}^{\mu-1} (x - x_k)$ solves the problem, in the second $\frac{1}{r^*} \prod_{k=0}^{\mu-1} (x - x_k)$ does.

If r exists, its canonical representation reads

$$r(x) = \frac{\sum_{k=0}^m a_k x^k}{\sum_{k=0}^n b_k x^k}$$

and the interpolation conditions (2) imply

$$p(x_k) - f_k q(x_k) = 0, \quad k = 0(1)N \quad (3)$$

or $a_0 + a_1 x_k + \dots + a_m x_k^m - f_k (b_0 + b_1 x_k + \dots + b_n x_k^n) = 0$. The set of solution vectors $[a_0 \ a_1 \ \dots \ a_m \ b_0 \ b_1 \ \dots \ b_n]^T$ of (3) is the kernel of the $(N + 1) \times (m + n + 2)$ -matrix

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^m & -f_0 & -f_0 x_0 & -f_0 x_0^2 & \cdots & -f_0 x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^m & -f_1 & -f_1 x_1 & -f_1 x_1^2 & \cdots & -f_1 x_1^n \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & x_N & x_N^2 & \cdots & x_N^m & -f_N & -f_N x_N & -f_N x_N^2 & \cdots & -f_N x_N^n \end{bmatrix}. \quad (4)$$

In order to have a nontrivial solution of (4) for every set of data, one should have less rows than columns. One therefore takes

$$N = m + n. \quad (5)$$

The kernel of (4) then always contains vectors for which $q \not\equiv 0$.

An introduction to classical rational interpolation can be found in [Bul-Rut], [Sto], [Wer-Scha] (the latter shortened in [Scha-Wer]). For further leading literature, the reader may consult [Gra1], [Mei], [Gut], [Wuy] and the literature cited there.

The main difficulty with classical rational interpolation are the zeros of q , of which we can distinguish two kinds:

- a) for the *zeros* z_* *common to* p one can (in theory) cancel the corresponding factors $x - z_*$. The kernel of (4) corresponds to a unique interpolant, *if the latter exists*. However, if a zero of q with multiplicity ν is a node x_k , then in view of (2) it is also a zero of p ; after cancellation of $(x - x_k)^\nu$, (1) takes a value which may be different from f_k : then the problem has no solution, the point (x_k, f_k) is called

unattainable. In view of the unicity theorem (Theorem 3 in [Wuy]), cancelling factors $x - z_*$ cannot introduce unattainable points. These are a property of the problem. Detecting them from the classical representation requires computing the values of q at all nodes.

- b) the *zeros not common to p* correspond to poles of r and can be classified into two kinds: whereas those lying outside the interval $[\min x_k, \max x_k]$ do not cause trouble, those inside it are the main drawback of rational interpolation. And the coefficients in the canonical representation of r do not give any hint to the presence of such poles.

2. Barycentric representation of the interpolant

Every rational interpolant $r \in \mathcal{R}_{N,N}$ can be written in its *barycentric form* [Ber-Mit]

$$r(x) = \sum_{k=0}^N \frac{u_k}{x - x_k} f_k \bigg/ \sum_{k=0}^N \frac{u_k}{x - x_k}. \quad (6)$$

Indeed, let $q_k := q(x_k)$ be the values of the denominator at the nodes; then

$$q(x) = \prod_{i=0}^N (x - x_i) \sum_{k=0}^N \frac{w_k}{x - x_k} q_k \quad (7)$$

with

$$w_k = 1 \bigg/ \prod_{i=0, i \neq k}^N (x_k - x_i) \quad (8)$$

is the Lagrangian representation of the denominator and r can be written as in (6) with

$$u_k := w_k q_k. \quad (9)$$

(This proof is an illustration of the fact, used by most constructive methods, that a rational interpolant is fully determined by its denominator.) Since $w_k \neq 0 \forall k$, there follows from (9) that q has a zero at a node iff the corresponding weight is itself zero.

There corresponds to every node x_k a so-called *weight* u_k , and a barycentric formula for r thus encompasses $N + 1$ unknowns, as opposed to $N + 2$ in a canonical representation.

The barycentric representation presents several advantages in comparison with the classical one:

- a) *unattainable points*: (6) implies that the interpolation condition at x_ℓ is satisfied for all $u_\ell \neq 0$:

$$\lim_{x \rightarrow x_\ell} \sum_{k=0}^n \frac{u_k}{x - x_k} f_k \bigg/ \sum_{k=0}^n \frac{u_k}{x - x_k} = f_\ell. \quad (10)$$

$u_\ell = 0$ therefore is a necessary condition for an unattainable point at x_ℓ : the barycentric weights give immediate information about possible unattainable points (see also [Ber2]); $u_\ell = 0$ in (6) simply means that the information at x_ℓ is discarded when determining the interpolant;

- b) *stability*: from (10) there follows that as long as $u_k \neq 0 \forall k$ the interpolation conditions are satisfied even if the weights are not those of the proper interpolant (i.e., if r in (6) does not have the right numerator and/or denominator degree(s)): the barycentric representation therefore is perfectly stable as far as the interpolation conditions are concerned;

c) *poles in* $[\min x_k, \max x_k]$: one has the following theorem [Sch-Wer]:

Theorem 2.1

Suppose the nodes are ordered as $x_0 < x_1 < \dots < x_N$, the common factors in r have been simplified to yield the reduced function \tilde{r} so that (6) corresponds to an interpolant with minimal denominator degree, and suppose $u_k \neq 0 \forall k$. Then $\text{sign } u_{k+1} = \text{sign } u_k$ implies that \tilde{r} has an odd number of poles in $[x_k, x_{k+1}]$.

Non-alternating signs of the weights of a reduced interpolant therefore is a sufficient condition for the presence of poles. Unfortunately, this condition is not necessary [Ber2]; finding necessary conditions deserves further research efforts.

3. Matrices for the determination of the weights

The only published algorithm for computing the weights u_k seems to be the one advocated by Schneider and Werner [Sch-Wer], who suggest finding first the Newton form of the denominator q ; more precisely, they use the vanishing of finite differences of $f q$:

$$f q[x_0, x_1, \dots, x_m, x_i] = 0, \quad i = m + 1(1)n.$$

Writing q in its Newton form $q(x) = \sum_{i=0}^n v_i \prod_{j=0}^{i-1} (x - x_j)$, this yields the homogeneous system with matrix

$$\left[\begin{array}{cccc} f[x_0, x_1, \dots, x_m, x_{m+1}] & f[x_1, x_2, \dots, x_m, x_{m+1}] & \dots & f[x_n, \dots, x_m, x_{m+1}] \\ \vdots & \vdots & & \vdots \\ f[x_0, x_1, \dots, x_m, x_{m+n}] & f[x_1, x_2, \dots, x_m, x_{m+n}] & \dots & f[x_n, \dots, x_m, x_{m+n}] \end{array} \right] \in \mathbb{R}^{n, n+1} \quad (11)$$

for the vector $\mathbf{v} := [v_0, v_1, \dots, v_n] \in \mathbb{R}^{n+1}$. This vector of the Newton coefficients of q is then transformed into a vector \mathbf{u} of the Lagrangian form of q by an algorithm of Werner [Wer].

We will present here a direct method for determining the vector $\mathbf{u} := [u_0, u_1, \dots, u_N]^T$. In view of the fact that, by (10), the barycentric form automatically guarantees interpolation, all we must do is achieve that the denominator and numerator degrees do not exceed n , respectively m . For that purpose, let

$$\ell(x) := (x - x_0)(x - x_1) \dots (x - x_N) = \prod_{j=0}^N (x - x_j)$$

denote the product in the Lagrangian representation of polynomial interpolation, as in (7), and start with the canonical representation of q :

$$q(x) = b_0 + b_1 x + \dots + b_{N-1} x^{N-1} + b_N x^N. \quad (12)$$

Then $x^N q\left(\frac{1}{x}\right) = b_0 x^N + b_1 x^{N-1} + \dots + b_{N-1} x + b_N$ and $q(x)$ is of degree $\leq N - 1$ iff

$$b_N = \lim_{x \rightarrow 0} x^N q\left(\frac{1}{x}\right) = 0.$$

Using (7) in order to translate this into a condition for the u_k 's, we get

$$\begin{aligned} x^N q\left(\frac{1}{x}\right) &= x^N \ell\left(\frac{1}{x}\right) \sum_{k=0}^N \frac{u_k}{\frac{1}{x} - x_k} \\ &= x^N \prod_{j=0}^N \left(\frac{1 - x_j x}{x}\right) \sum_{k=0}^N \frac{u_k}{\frac{1 - x_k x}{x}} \\ &= x^N \frac{1}{x^{N+1}} \left[\prod_{j=0}^N (1 - x_j x) \right] x \sum_{k=0}^N \frac{u_k}{1 - x_k x}, \end{aligned}$$

so that

$$b_N = \lim_{x \mapsto 0} q\left(\frac{1}{x}\right) x^N = \sum_{k=0}^N u_k$$

and

$$\deg q \leq N - 1 \iff b_N = \sum_{k=0}^N u_k = 0. \quad (13)$$

The replacement of u_k by $u_k f_k$ is the only difference between q and p , and so

$$\deg p \leq N - 1 \iff \sum_{k=0}^N u_k f_k = 0.$$

Once $b_N = 0$ is satisfied, (12) shows as above that $b_{N-1} = 0$ iff

$$\lim_{x \mapsto 0} x^{N-1} q\left(\frac{1}{x}\right) = 0. \quad (14)$$

Written in terms of the u_k 's, the quantity whose limit is sought reads

$$x^{N-1} q\left(\frac{1}{x}\right) = \frac{1}{x} \left[\prod_{j=0}^N (1 - x_j x) \right] \sum_{k=0}^N \frac{u_k}{1 - x_k x}.$$

As we want to let $x \mapsto 0$, each term of the last sum can be expanded into its geometric series

$$\frac{u_k}{1 - x_k x} = u_k (1 + x_k x + (x_k x)^2 + \dots)$$

and the sum becomes

$$\sum_{k=0}^N \frac{u_k}{1 - x_k x} = \sum_{k=0}^N u_k + x \sum_{k=0}^N u_k x_k + x^2 \left(\sum_{k=0}^N u_k x_k^2 + \dots \right).$$

Using (13) and letting $x \mapsto 0$, we see that (14) and the corresponding condition for p mean

$$\deg q \leq N - 2 \iff b_{N-1} = 0 \iff \sum_{k=0}^N u_k x_k = 0 \quad (15)$$

$$\deg p \leq N - 2 \iff \sum_{k=0}^N u_k f_k x_k = 0.$$

One can keep on decreasing the degrees of the denominator and the numerator in the same manner and prove the following by induction.

Theorem 3.1

Let $r = \frac{p}{q}$ be a rational function written in its barycentric form (6) with

$$p(x) = \ell(x) \sum_{k=0}^N \frac{u_k}{x - x_k} f_k, \quad q(x) = \ell(x) \sum_{k=0}^N \frac{u_k}{x - x_k}.$$

Then

$$\deg q \leq n \iff \sum_{k=0}^N x_k^i u_k = 0, \quad i = 0(1)N - (n + 1), \quad (16a)$$

$$\deg p \leq m \iff \sum_{k=0}^N f_k x_k^i u_k = 0, \quad i = 0(1)N - (m + 1). \quad (16b)$$

By choosing m and n satisfying (5) one sees that the set of \mathbf{u} 's that correspond to r solving the rational interpolation problem is the kernel of the $N \times (N + 1)$ -matrix

$$\mathbf{A} := \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_0 & x_1 & x_2 & \cdots & x_N \\ x_0^2 & x_1^2 & x_2^2 & \cdots & x_N^2 \\ \vdots & \vdots & \vdots & & \vdots \\ x_0^{m-1} & x_1^{m-1} & x_2^{m-1} & \cdots & x_N^{m-1} \\ f_0 & f_1 & f_2 & \cdots & f_N \\ f_0 x_0 & f_1 x_1 & f_2 x_2 & \cdots & f_N x_N \\ f_0 x_0^2 & f_1 x_1^2 & f_2 x_2^2 & \cdots & f_N x_N^2 \\ \vdots & \vdots & \vdots & & \vdots \\ f_0 x_0^{n-1} & f_1 x_1^{n-1} & f_2 x_2^{n-1} & \cdots & f_N x_N^{n-1} \end{bmatrix}. \quad (17)$$

Remarks:

- 1) The number of rows corresponding to the degree conditions for the denominator is equal to the degree of the numerator, and conversely;
- 2) the matrix (17) is not quite the transposed of (4): not only the negative signs (which would not affect the kernel) are missing, but also the dimensions are different as (4) has one more row and one more column;
- 3) in [Gra2] appeared a proof, attributed to Meinguet, of a system equivalent to (16): replacing there in (9) $\frac{1}{\ell'(x_k)}$ by w_k ([Hen] p. 243), using our (9), transposing and reordering the equations yields (16). \odot

4. Determination of the barycentric weights through triangulation of \mathbf{A}

In order to determine from (16) the weights of rational interpolation, i.e., the kernel of \mathbf{A} in (17), we will now triangulate \mathbf{A} . For that purpose, we will call the first m rows (i.e., those without f_k 's) of \mathbf{A} its "top part" and the last n rows its "bottom part".

The triangulation will be performed in several steps:

1) *Separate triangulation of top part and bottom part:*

a) subtract x_N times each row from the next. With the space saving abbreviation

$$x_{jk} := x_{j,k} := x_j - x_k$$

this yields

$$\begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ x_{0N} & x_{1N} & \cdots & x_{N-1,N} & 0 \\ x_0 x_{0N} & x_1 x_{1N} & \cdots & x_{N-1} x_{N-1,N} & 0 \\ x_0^2 x_{0N} & x_1^2 x_{1N} & \cdots & x_{N-1}^2 x_{N-1,N} & 0 \\ \vdots & \vdots & & \vdots & 0 \\ x_0^{m-2} x_{0N} & x_1^{m-2} x_{1N} & \cdots & x_{N-1}^{m-2} x_{N-1,N} & 0 \\ f_0 & f_1 & \cdots & f_{N-1} & f_N \\ f_0 x_{0N} & f_1 x_{1N} & \cdots & f_{N-1} x_{N-1,N} & 0 \\ f_0 x_0 x_{0N} & f_1 x_1 x_{1N} & \cdots & f_{N-1} x_{N-1} x_{N-1,N} & 0 \\ f_0 x_0^2 x_{0N} & f_1 x_1^2 x_{1N} & \cdots & f_{N-1} x_{N-1}^2 x_{N-1,N} & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ f_0 x_0^{n-2} x_{0N} & f_1 x_1^{n-2} x_{1N} & \cdots & f_{N-1} x_{N-1}^{n-2} x_{N-1,N} & 0 \end{bmatrix}$$

b) to obtain the zeros in the second column, subtract x_{N-1} times each of the rows number 2 to m , respectively n , from the next to get

$$\begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 1 \\ x_{0N} & x_{1N} & \cdots & x_{N-2,N} & x_{N-1,N} & 0 \\ x_0 x_{0,N-1} x_{0N} & x_1 x_{1,N-1} x_{1N} & \cdots & x_{N-2} x_{N-2,N-1} x_{N-2,N} & 0 & 0 \\ x_0^2 x_{0,N-1} x_{0N} & x_1^2 x_{1,N-1} x_{1N} & \cdots & x_{N-2}^2 x_{N-2,N-1} x_{N-2,N} & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ x_0^{m-3} x_{0,N-1} x_{0N} & x_1^{m-3} x_{1,N-1} x_{1N} & \cdots & x_{N-2}^{m-3} x_{N-2,N-1} x_{N-2,N} & 0 & 0 \\ f_0 & f_1 & \cdots & f_{N-2} & f_{N-1} & f_N \\ f_0 x_{0N} & f_1 x_{1N} & \cdots & f_{N-2} x_{N-2,N} & f_{N-1} x_{N-1,N} & 0 \\ f_0 x_0 x_{0,N-1} x_{0N} & f_1 x_1 x_{1,N-1} x_{1N} & \cdots & f_{N-2} x_{N-2} x_{N-2,N-1} x_{N-2,N} & 0 & 0 \\ f_0^2 x_0^2 x_{0,N-1} x_{0N} & f_1^2 x_1^2 x_{1,N-1} x_{1N} & \cdots & f_{N-2}^2 x_{N-2}^2 x_{N-2,N-1} x_{N-2,N} & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ f_0 x_0^{n-3} x_{0,N-1} x_{0N} & f_1 x_1^{n-3} x_{1,N-1} x_{1N} & \cdots & f_{N-2} x_{N-2}^{n-3} x_{N-2,N-1} x_{N-2,N} & 0 & 0 \end{bmatrix}$$

c) pursue in the same way, until both matrices are (separately) triangulated:

$$\begin{bmatrix} 1 & \cdots & & 1 & & 1 & 1 \\ x_{0N} & \cdots & & x_{N-2,N} & & x_{N-1,N} & \\ x_0 x_{0,N-1} x_{0N} & \cdots & & x_{N-2} x_{N-2,N-1} x_{N-2,N} & & & \\ \vdots & & \ddots & & & & \\ x_0 x_{0,n+2} \cdots x_{0N} & \cdots & x_{n+1,n+2} \cdots x_{n+1,N} & & & & \\ f_0 & \cdots & & \cdots & & f_{N-1} & f_N \\ f_0 x_{0N} & \cdots & & & & f_{N-1} x_{N-1,N} & \\ \vdots & & & & & & \\ f_0 x_0 x_{0,m+2} \cdots x_{0N} & \cdots & & f_{m+1} x_{m+1,m+2} \cdots x_{m+1,N} & & & \end{bmatrix}. \quad (18)$$

The elimination in the top part is now complete.

Remarks:

α) Since the order of the nodes is arbitrary, the last row of the top matrix implies that, given any subset S of $n+2$ nodes $x_{i_0}, x_{i_1}, \dots, x_{i_{n+1}}$ with corresponding weights $u_{i_0}, u_{i_1}, \dots, u_{i_{n+1}}$, one has

$$\sum_{k=0}^{n+1} d_{i_k} u_{i_k} = 0 \quad (19a)$$

where

$$d_{i_k} := (x_{i_k} - x_{i_{n+2}})(x_{i_k} - x_{i_{n+3}}) \cdots (x_{i_k} - x_{i_N}) \quad (19b)$$

denotes the product of the differences between x_{i_k} and all x_{i_ℓ} , $i_\ell \notin S$. Moreover, if one applies Gauss-Jordan elimination to the top matrix in (18) in the same manner as above, but here by subtracting a multiple of the *next* row to annulate an element, and multiplies by the arising denominators, a “lower anti-triangular” matrix with $n+2$ anti-diagonals results whose rows contain the coefficients in (19) for the S with $n+2$ consecutive indices;

β) since all d_{i_k} are different from 0, the above remark implies that the top matrix has full rank;

γ) in the special case of *polynomial interpolation* ($n = 0$), a full-rank bidiagonal matrix results, whose last row implies

$$u_1 = -\frac{x_{02}x_{03}\dots x_{0N}}{x_{12}x_{13}\dots x_{1N}}u_0 = -\frac{\prod_{i=2}^N(x_0 - x_i)}{\prod_{i=2}^N(x_1 - x_i)}u_0.$$

Changing the order of the variables to bring x_k into the second column (or solving the bidiagonal system by back substitution) similarly yields

$$u_k = \frac{\prod_{i=1, i \neq k}^N(x_0 - x_i)}{\prod_{i=1, i \neq k}^N(x_k - x_i)}u_0.$$

Choosing $u_0 = w_0 = 1 / \prod_{i=1}^N(x_0 - x_i)$ for the arbitrary u_0 , one obtains $u_k = w_k$ from (8), the barycentric weights of polynomial interpolation. As a corollary we get the thus far possibly overlooked fact that *the kernel of a transposed Vandermonde matrix without its last row is the space of the barycentric weights of polynomial interpolation between the points making up the Vandermonde.* \odot

We now continue our triangulation of \mathbf{A} :

2) *Elimination of the diagonals of the bottom matrix present in the top matrix:*

a) subtract successively f_{N-k+1} times the k -th row of the top from the k -th of the bottom, $k = 1(1)n = N - m$, and express the differences of function values with finite differences, i.e., $f_j - f_k = (x_j - x_k)f[x_j, x_k]$.

This yields

$$\left[\begin{array}{cccccccc} x_{0N}f[x_0, x_N] & \dots & x_{m,N}f[x_m, x_N] & \dots & \dots & \dots & x_{N-1,N}f[x_{N-1}, x_N] & 0 \\ x_{0,N-1}x_{0N}f[x_0, x_{N-1}] & \dots & x_{m,N-1}x_{m,N}f[x_m, x_{N-1}] & \dots & x_{N-2,N-1}x_{N-2,N}f[x_{N-2}, x_{N-1}] & 0 & 0 & 0 \\ \vdots & & \vdots & & & & \ddots & \\ \vdots & & \vdots & & & & & \\ x_{0,m+1}\dots x_{0N}f[x_0, x_{m+1}] & \dots & x_{m,m+1}\dots x_{mN}f[x_m, x_{m+1}] & & & & & \end{array} \right]$$

b) in order to eliminate the next diagonal, subtract successively $f[x_{N-k+1}, x_{N-k+2}]$ times the k -th row of the top from the k -th of the bottom, $k = 2(1)n + 1 = N - m + 1$, to get

$$\left[\begin{array}{cccccccc} x_{0,N-1}x_{0N}f[x_0, x_{N-1}, x_N] & \dots & x_{m-1,N-1}x_{m-1,N}f[x_{m-1}, x_{N-1}, x_N] & \dots & x_{N-2,N-1}x_{N-2,N}f[x_{N-2}, x_{N-1}, x_N] & 0 & 0 & 0 \\ x_{0,N-2}x_{0,N-1}x_{0N}f[x_0, x_{N-2}, x_{N-1}] & \dots & x_{m-2,N-2}x_{m-2,N-1}x_{m-2,N}f[x_{m-2}, x_{N-2}, x_{N-1}] & \dots & x_{N-3,N-2}x_{N-3,N-1}x_{N-3,N}f[x_{N-3}, x_{N-2}, x_{N-1}] & 0 & 0 & 0 \\ \vdots & & \vdots & & \ddots & & & \\ \vdots & & \vdots & & & & & \\ x_{0,m}\dots x_{0N}f[x_0, x_m, x_{m+1}] & \dots & x_{m-1,m}\dots x_{mN}f[x_{m-1}, x_m, x_{m+1}] & & & & & \end{array} \right]$$

c) ... and so on. For each eliminated diagonal, the number of factors of differences of x -values as well as the number of arguments in the divided differences increase by one. After eliminating $m - n + 1$ diagonals, one has (without the columns of zeros)

$$\left[\begin{array}{cccc} x_{0,2n}\dots x_{0N}f[x_0, x_{2n}, \dots, x_N] & \dots & x_{n,2n}\dots x_{nN}f[x_n, x_{2n}, \dots, x_N] & \dots & x_{2n-1,2n}\dots x_{2n-1,N}f[x_{2n-1}, x_{2n}, \dots, x_N] \\ \vdots & & \vdots & & \ddots \\ x_{0,n+1}\dots x_{0N}f[x_0, x_{n+1}, \dots, x_{m+1}] & \dots & x_{n,n+1}\dots x_{nN}f[x_n, x_{n+1}, \dots, x_{m+1}] & & \end{array} \right]$$

The number of factors of differences of x -values increases by one from row to row, the numbers of arguments in the divided differences is $m - n + 2$ everywhere.

3) eliminate similarly the “triangle” on the right of the bottom matrix, i.e., the diagonals shortened at each step of one more element at their lower extremity. At the end the products of the differences of x_k ’s all have the same number of factors and are identical in every column. To save space again, they will be denoted in the following by

$$X_k := x_{k,n+1}x_{k,n+2} \dots x_{k,N} = (x_k - x_{n+1})(x_k - x_{n+2}) \dots (x_k - x_N), \quad k = 0(1)n.$$

The finite differences have a number of arguments decreasing from the last to the first row. We inverse the order and get the $n \times (n + 1)$ -matrix

$$\begin{bmatrix} X_0f[x_0, x_{n+1}, \dots, x_{m+1}] & X_1f[x_1, x_{n+1}, \dots, x_{m+1}] & \dots & X_nf[x_n, x_{n+1}, \dots, x_{m+1}] \\ X_0f[x_0, x_{n+1}, \dots, x_{m+2}] & X_1f[x_1, x_{n+1}, \dots, x_{m+2}] & \dots & X_nf[x_n, x_{n+1}, \dots, x_{m+2}] \\ \vdots & \vdots & \dots & \vdots \\ X_0f[x_0, x_{n+1}, \dots, x_{N-1}] & X_1f[x_1, x_{n+1}, \dots, x_{N-1}] & \dots & X_nf[x_n, x_{n+1}, \dots, x_{N-1}] \\ X_0f[x_0, x_{n+1}, \dots, x_N] & X_1f[x_1, x_{n+1}, \dots, x_N] & \dots & X_nf[x_n, x_{n+1}, \dots, x_N] \end{bmatrix}, \quad (20)$$

whose kernel is the space of the first $n + 1$ barycentric weights (the m last ones are then computed from the top matrix).

Remarks:

- δ) notice the difference with the matrix (11): whereas there the number of arguments of the divided differences changes from column to column, in (20) it changes from row to row;
- ε) *special case: denominator of degree 1* ($n = 1$). The only row of (20) then reads

$$(x_0 - x_2)(x_0 - x_3) \dots (x_0 - x_N)f[x_0, x_2, \dots, x_N]u_0 + (x_1 - x_2)(x_1 - x_3) \dots (x_1 - x_N)f[x_1, x_2, \dots, x_N]u_1 = 0$$

and since, as noticed in remark α), the order of the nodes is arbitrary, we have

$$u_k = - \frac{f[x_0, \{x_\ell, \ell \neq 0, k\}] \prod_{\ell \neq 0, k} (x_0 - x_\ell)}{f[x_k, \{x_\ell, \ell \neq 0, k\}] \prod_{\ell \neq 0, k} (x_k - x_\ell)} u_0.$$

Moreover, if the two elements of (20) are identical, then by the last formula u_1 is related to u_0 as w_1 to w_0 , and by (19) the coefficients are the w_k in (8): r is the interpolating polynomial;

- ζ) from one column to the next, the arguments of the divided differences differ only by a single argument. The common differences $f[x_{n+1}, \dots, x_{m+1}]$, $f[x_{n+1}, \dots, x_{m+2}]$, \dots , $f[x_{n+1}, \dots, x_N]$ can be computed first in about $\frac{m^2}{2}$ flops in a single difference tableau. Then this same tableau is updated for each of the nodes x_0, \dots, x_n , which gives the differences in the columns of (20) one after another. The cost of computing all divided differences thus is about $\frac{m^2}{2} + m(n + 1)$ flops;
- η) the divided differences need not be multiplied by X_0, \dots, X_n : these factors can be taken care of at the final back substitution (see below);
- ζ) up till here, the elimination is linear in the values f ; the nonlinearity comes into play in the next, last step. ◊

4) triangulation of (20) by *Gaussian “anti”-elimination*. The node x_ℓ corresponding to the variable u_ℓ eliminated from a row comes into the other elements of the row, so that in the final bottom part the

number of arguments in the elements increases by two from one row to the next. We will denote by $F[\quad]$ the elements computed in the process (without the X_k 's), starting with $F[\quad] := f[\quad]$. Wenn x_ℓ is eliminated, the k -th element of the j -th row changes according to the following program:

```

for  $\ell := n(-1)2$  do
  for  $j := N - \ell + 2(1)N$  do
    for  $k := \ell - 1(-1)0$  do

```

$$F[x_k, x_\ell, \dots, x_j] = F[x_k, x_{\ell+1}, \dots, x_j] - \frac{F[x_\ell, \dots, x_j]}{F[x_\ell, \dots, x_{N-\ell+1}]} F[x_k, x_{\ell+1}, \dots, x_{N-\ell+1}] \quad (21)$$

```

end  $k$ ; end  $j$ ; end  $\ell$ .

```

The bottom matrix finally reads

$$\left[\begin{array}{cccc} X_0 f[x_0, x_{n+1}, \dots, x_{m+1}] & X_1 f[x_1, x_{n+1}, \dots, x_{m+1}] & \dots & X_{n-1} f[x_{n-1}, x_{n+1}, \dots, x_{m+1}] & X_{n-1} f[x_n, x_{n+1}, \dots, x_{m+1}] \\ X_0 F[x_0, x_n, \dots, x_{m+2}] & X_1 F[x_1, x_n, \dots, x_{m+2}] & \dots & X_{n-1} F[x_{n-1}, x_n, \dots, x_{m+2}] & 0 \\ \vdots & & \ddots & & \ddots \\ X_0 F[x_0, x_3, \dots, x_{N-1}] & & X_2 F[x_2, x_3, \dots, x_{N-1}] & & 0 \\ X_0 F[x_0, x_2, \dots, x_N] & X_1 F[x_1, x_2, \dots, x_N] & & & 0 \end{array} \right] \quad (22)$$

(21) requires that all pivots $F[x_\ell, \dots, x_{N-\ell+1}]$ are $\neq 0$. The case in which the latter is true (also for $\ell = 1$) is the generic case.

The method unfortunately requires $\mathcal{O}(n^3)$ flops. The challenge would be to compute the F in $\mathcal{O}(n^2)$ operations.

Now that the matrix is triangulated, one can find \mathbf{u} by back-substitution: starting from any nonvanishing value for u_0 ($u_0 = 0$ yields the uninteresting trivial solution $\mathbf{u} = \mathbf{0}$), one successively gets u_1, u_2, \dots, u_n by (22), then u_{n+1}, \dots, u_N by the “ $(n+2)$ -lower antidiagonal” (19). The latter is preferable to the triangular top part of (18), since the number of additions/subtractions is smaller. (In particular, the first row of (18) is to be avoided: it would give the last coefficient as $u_N = -\sum_{k=0}^{N-1} u_k$, a long sum with mainly alternating signs, thus subject to catastrophic cancellation and smearing [Hen].)

5. Implementation issues and non-generic cases

We add here practical remarks on the use of the above method:

- 1) Computing divided differences as those needed in (20) for nodes ordered from one side to the other of an interval is a notoriously unstable process: the points should be re-ordered, e.g. with the van der Corput sequence [Fis-Rei,Tal]; the order of the weights must be corrected accordingly once back-substitution is complete.
- 2) Even in the generic case, if one wants to avoid exaggerated growth of the $F[\quad]$ in (21), pivoting should be used. We used column pivoting in order to keep the natural order of the degree conditions. For the weights, this means a change of their order, which must be stored.
- 3) If a row is totally zero (no pivot), the kernel is at least two-dimensional, the solution is not anymore unique (up to a constant) and the problem therefore ill-posed. There are at least two ways of coping with this:

- if one is interested in the general solution, one should keep the corresponding weight indeterminate and compute the kernel as a function of all the undeterminate weights, a usual way of finding kernels;
- instead, we have modified the problem to make it well-posed: with the desire to hold the number of poles as low as possible, we have decreased n and increased m accordingly until the solution was unique. Therefore the problem we have solved should be rephrased as follows: *find the unique $r \in \mathcal{R}_{m^*, n^*}$ with $m^* + n^* = N$, $n^* \leq m^*$, that satisfies the interpolation conditions (2) with $n^* \leq n$ as large as possible.* Since the solution is then one-dimensional, the rational function is reduced. In that sense, we have determined (if it exists) the interpolant with minimal denominator degree as in [Sch-Wer] and [Wuy]. Row pivoting would theoretically allow the immediate determination of n^* .

In the top matrix, decreasing n and increasing m means erasing a factor from every d_{ik} in (19b) and computing one more row. In the bottom (before triangulation), this means cancelling the first row and column and update the remaining divided differences with x_n . Similar remarks yield for the updating problem of increasing N , to which we did not give much thought, however.

Contraposition of Theorem 2.1 can be useful in determining whether a row is zero or not (i.e., whether or not the kernel has dimension one):

Corollary 5.1

Suppose the nodes are ordered as $x_0 < x_1 < \dots < x_N$ and $u_k \neq 0 \forall k$. Then if $\text{sign } u_k = \text{sign } u_{k+1}$ for a k and if r is bounded on $[x_0, x_N]$, then it is not reduced.

- 4) When one u_ℓ is zero, then, since there is one less term, (13) already implies that $\deg q \leq N - 2$, (15) that $\deg q \leq N - 3$, etc. Numerator and denominator then automatically have degree 1 less than required. In fact, $u_\ell = 0$ in (16) implies that the factor $x - x_\ell$ has been cancelled in the numerator and the denominator, and $p(x_\ell) = q(x_\ell) = 0$;
- 5) Numerical experience leads us to the following two conjectures:
 - even with column pivoting, the rows of zeros are at the end of the triangulated matrix;
 - if the nodes lie symmetrically with respect to the center of $[\min x_k, \max x_k]$, then the weights are symmetric, even for a function that does not display any symmetry.
- 6) With the reduced r , one can use the criteria of §2 for detecting unattainable points and poles:
 - if $u_\ell = 0$, evaluate r at x_ℓ : if it is different from f_ℓ , (x_ℓ, f_ℓ) is unattainable; we did not encounter any example where our method gave $u_\ell = 0$ and x_ℓ was not unattainable, and therefore conjecture that the method yields $u_\ell = 0$ iff x_ℓ is unattainable (see also Corollary 7 in [Sch-Wer], which seems to lack some hypothesis to be true as generally as stated);
 - same signs of weights corresponding to consecutive nodes guarantee that r has an odd number of poles between these.

6. Numerical examples

We have tested the method described above on dozen of examples with MATLAB for the MacIntosh. First we have recomputed the low-dimensional examples on pp. 293–294 of [Sch-Wer] and obtained the same coefficients in all examples; and in example c) they came out directly, without the detour by a random Newton denominator.

In order to demonstrate what happens with unattainable points, let us try the example $\mathbf{x} := [-2, -1, 0, 1, 2]^T$, $\mathbf{y} := [1, 2, -1, 0, 1]^T$, $m = n = 2$. Here the matrix \mathbf{A} in (17) has (full) rank 4, its one-dimensional kernel is spanned by the weights $\mathbf{u} = [1, -1, -1, 1, 0]^T$ corresponding (without reduction) to the interpolant $r(x) = (x-1)/(2x+1)$ with $r(2) = 1/5$: the point (2,1) is unattainable. The weights of identical signs in -1 and 0 reflect the pole in $-1/2$. We mention that with the first 4 points and $m = 2$, $n = 1$ we get the same \mathbf{u} without the zero value. With $\hat{\mathbf{y}} := [1, 2, -1, 0, 1/5]^T$ instead of \mathbf{y} as input one gets indeed again $r(x)$, but \mathbf{A} has rank 3 (last row is zero after triangulation). Within the two-dimensional kernel the method chooses $\hat{\mathbf{u}} = [1, 1/3, 3, -11, 20/3]^T$. Without reduction this corresponds to $\hat{r}(x) = (5x^2 + x - 6)/(10x^2 + 17x + 6)$. The sign pattern reflects the zeros of the denominator at $x = 6/5$ and $x = -1/2$, and not to the poles of \hat{r} which reduces to r . We have not been able (lack of patience?) to construct an example where the method still chooses after correction of \mathbf{y} to $\hat{\mathbf{y}}$ a $\hat{\mathbf{u}}'$ with the same zero component and no unattainable point at the corresponding abscissa.

According to remark 3), the zero row leads us to set $m = 3$, $n = 1$. Then the method yields the unique solution $\hat{\hat{\mathbf{u}}} = [1, -4/3, -2, 4, -5/3]^T$, which corresponds to r (without reduction) and again reflects the pole at $x = -1/2$.

Next we have tried a modified Bulirsch-Rutishauser example [Bul-Rut, p.288]: interpolate $f(x) = \cot x$ between equidistant points on $[0.5^0, 5^0]$, and compare at $x = 1.5^0$ the value of r with that of f . In view of the simple pole of f at $x = 0$, for any given N the interpolant is about as good with $n = 1$ as with any larger n . $N = 7$ is sufficient for about machine precision, and the latter is conserved up to about $N = 40$. Then the precision gradually decreases because of smearing: for $N = 100$ the error is $7.5 \cdot 10^{-9}$. With the interpolating polynomial in barycentric form (see (7) and (8) or [Hen, p.238]), $N \approx 40$ is necessary for machine precision. And at $x = 0.75$, thus closer to the extremity of the interval and the pole, $r(0.75)$ with $N = 7$, $n = 1$, still has machine precision, whereas the error with the polynomial decreases to $5.2 \cdot 10^{-9}$ for $N = 40$ before increasing again because of smearing.

If one chooses $n = 2$, the two nonzero entries of the second line of (22), which have very similar sizes, decrease with N : for $N = 5$, they are $-3.6 \cdot 10^{-7}$ and $-4.6 \cdot 10^{-7}$, for $N = 8$, $-6.4 \cdot 10^{-13}$ and $-7.0 \cdot 10^{-13}$, and the u_k 's have alternating signs. The maximal error $\|r - f\|$, as measured by the maximal value of $|r(x) - f(x)|$ at 109 equidistant x between 0.5^0 and 5^0 , is $3.7 \cdot 10^{-13}$. With $N = 9$, the entries of the second line of (22) become $-4.8 \cdot 10^{-15}$ and $1.5 \cdot 10^{-15}$. Is the kernel two-dimensional? Corollary 5.1 gives the answer: u_5 and u_6 have the same signs and $\|r - f\| = 7.1 \cdot 10^{-14}$, thus r has no pole on $[0.5^0, 5^0]$: the kernel is two-dimensional. However, this does not work for $N = 10$: the entries of (22) are $2.7 \cdot 10^{-15}$ and $1.6 \cdot 10^{-15}$, the signs of the u_k 's alternate and $\|r - f\| = 2.4 \cdot 10^{-11}$. But it works again for $11 \leq N \leq 17$, and we conjecture that it does for larger N .

Finally we have experimented with $f(x) = e^{-(x+1.2)}/(1 + 25x^2)$, first between equidistant points on $[-1, 1]$. f displays an essential singularity at -1.2 and Runge's phaenomenon makes it completely unsuitable to polynomial interpolation close to the extremities of the interval of interpolation. For that reason we have evaluated r at $x = -0.95$, where the rounded exact value is 2.31716286612814. Because of the essential singularity, the best approximation results are obtained for most N with n as large as possible, i.e., $n = N/2$ for N even and $n = (N-1)/2$ for N odd. The results are displayed in Table 1. They show that the method works very well, the only difficulty being again smearing when N becomes larger than about 30.

To allow comparison with the polynomial, we have interpolated also between Čebyšev points of the second kind on $[-1, 1]$, $x_k = \cos k \frac{\pi}{N}$. The barycentric weights were given in [Sal], see [Ber1] for an alternative treatment. The third column of Table 2 contains the results (the value of n is relevant only for the rational:

Table 1

Errors at $x = -0.95$ with rational interpolation of $f(x) = e^{-(x+1.2)}/(1+25x^2)$ between equidistant points on $[-1, 1]$

| N | n | Error |
|-----|-----|-------------|
| 3 | 1 | $3.19e - 3$ |
| 7 | 3 | $5.35e - 1$ |
| 15 | 7 | $3.46e - 6$ |
| 31 | 15 | $1.1e - 10$ |
| 63 | 31 | $5.77e - 8$ |

Table 2

Comparison of rational and polynomial errors when interpolating $f(x) = e^{-(x+1.2)}/(1+25x^2)$ between Čebyšev points on $[-1, 1]$

| N | n | $x = -0.95$ | | $x = -0.05$ | |
|-----|-----|-------------|-------------|-------------|-------------|
| | | Rational | Polynomial | Rational | Polynomial |
| 3 | 1 | 2.64 | 2.62 | 1.8 | 2.67 |
| 7 | 3 | $1.74e - 1$ | $6.47e - 1$ | $4.25e - 1$ | 1.18 |
| 15 | 7 | $8.19e - 8$ | $4.84e - 2$ | $8.4e - 13$ | $1.70e - 1$ |
| 31 | 15 | $5.2e - 14$ | $1.05e - 4$ | 0.0 | $1.82e - 4$ |
| 63 | 31 | 0.0 | $3.20e - 7$ | $2.7e - 15$ | $1.56e - 5$ |

the polynomial obviously corresponds to $n = 0$). Comparison with Table 1 shows that for this function rational interpolation between equidistant points is even better than polynomial interpolation between Čebyšev points! The table contains also the error at a point close to the center of the interval, where the Čebyšev points are not as dense as in the vicinity of the extremities. In this example our method was about 2 digits more precise than the alternative consisting in letting $b_0 := 1$ and solving the system (3) (the usual way, i.e., with partial pivoting) for the a_k 's and the remaining b_k 's.

Conclusions

The method given above for a direct computation of the barycentric representation of rational interpolants as the kernel of the matrix (17) seems very effective, at least as long as N is not too large. For very large N and difficult points like equidistant ones, higher precision should be used to cope with smearing. The method gives quite a precise information about the size of the solution space.

The barycentric representation should often be favored in view of the valuable information it gives about unattainable points and poles of the interpolant. One could surely think of determining the interpolant in more traditional ways, and in a second stage compute its barycentric weights. It is known, however, that the process of computing the barycentric weights of a rational interpolant from its canonical representation can be ill-conditioned [Hen p.236].

References

- [Ber1] Berrut J.-P., Baryzentrische Formeln zur trigonometrischen Interpolation (I), *Z. angew. Math. Phys. (ZAMP)* **35** (1984) 91–105.
- [Ber2] Berrut J.-P., Linear rational interpolation of continuous functions over an interval, in: W. Gautschi, ed., *Mathematics of Computation 1943–1993: a Half-Century of Computational Mathematics* (Proceedings of Symposia in Applied Mathematics, American Mathematical Society, Providence, 1994) 261–264.
- [Ber-Mit] Berrut J.-P., Mittelmann H., *Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval*, Report 94-4, Institut de Mathématiques, Université de Fribourg (Suisse), 1994.
- [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.
- [Fis-Rei] Fischer B., Reichel L., Newton interpolation in Fejér and Chebyshev points, *Math. Comp.* **53** (1989) 265–278.
- [Gra1] Graves–Morris P. R., Efficient reliable rational interpolation, in: M. G. de Gruin and H. van Rossum, eds., *Padé Approximation and its Applications, Amsterdam 1980* (LNM 888, Springer–Verlag, Berlin–Heidelberg–New York, 1981) 28–63.
- [Gra2] Graves–Morris P. R., Symmetrical formulas for rational interpolants, *J. Comput. Appl. Math.* **10** (1984) 107–111.
- [Hen] Henrici P., *Essentials of Numerical Analysis* (John Wiley, New York, 1982).
- [Gut] Gutknecht M. H., Block structure and recursiveness in rational interpolation, in: Cheney E. W., Chui C. K. and L. L. Schumaker (eds.), *Approximation Theory VII* (Academic Press, Boston, 1992) 93–130.
- [Mei] Meinguet J., On the solubility of the Cauchy interpolation problem, in: Talbot A., ed., *Approximation Theory* (Academic Press, London and New York, 1970) 137–163.
- [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.
- [Scha-Wer] Schaback R., Werner H., *Numerische Mathematik* (Springer-Verlag, Berlin 1992).
- [Sch-Wer] Schneider C., Werner W., Some new aspects of rational interpolation, *Math. Comp.* **47** (1986) 285–299.
- [Sto] Stoer J., *Einführung in die Numerische Mathematik I* (4. Aufl., Springer, Berlin–Heidelberg–New York, 1983).
- [Tal] Tal–Ezer H., High degree polynomial interpolation in Newton form, *SIAM J. Sci. Stat. Comput.* **12** (1991) 648–667.
- [Wer-Scha] Werner H., Schaback R., *Praktische Mathematik II* (Springer-Verlag, Berlin, 1972).
- [Wer] Werner W., Polynomial interpolation: Lagrange versus Newton, *Math. Comp.* **43** (1984) 205–217.
- [Wuy] Wuytack L., On some aspects of the rational interpolation problem, *SIAM J. Numer. Anal.* **11** (1974) 52–60.