Algorithms for solving rational interpolation problems related to fast and superfast solvers for Toeplitz systems

Peter Kravanja and Marc Van Barel

Katholieke Universiteit Leuven, Department of Computer Science
Celestijnenlaan 200 A, B-3001 Heverlee, Belgium

ABSTRACT

Linearized rational interpolation problems at roots of unity play a crucial role in the fast and superfast Toeplitz solvers that we have developed. Our interpolation algorithm is a sequential algorithm in which a matrix polynomial that satisfies already some of the interpolation conditions is updated to satisfy two additional interpolation conditions. In the algorithm that we have used so far, the updating matrix, which is a matrix polynomial of degree one, is constructed in a two-step process that resembles Gaussian elimination. We briefly recall this approach and then consider two other approaches. The first one is a completely new approach based on an updating matrix that is unitary with respect to a discrete inner product that is based on roots of unity. The second one is an application of an algorithm for solving discrete least squares problems on the unit circle, a problem that has linearized rational interpolation at roots of unity as its limiting case. We conduct a number of numerical experiments to compare the three strategies.

Keywords: Toeplitz matrices, rational interpolation, fast and superfast algorithms, pivoting

1. INTRODUCTION

Rational interpolation problems play an important role in computational mathematics, in particular in numerical linear algebra, signal processing and system theory. The literature on the subject is vast. We will not give a comprehensive overview and instead refer the interested reader to the papers by Meinguet,[1] Antoulas,[2–4] Berrut and Mittelmann[5,6] or Gutknecht[7–10] and the references cited therein.

Our interest in rational interpolation is motivated by the fact that linearized rational interpolation problems at roots of unity play a crucial role in the fast and superfast algorithms for solving linear systems of equations that have Hankel or Toeplitz structure that we have developed. An algorithm is called "fast" if it solves a problem of size n in $\mathcal{O}(n^2)$ flops. A "superfast" algorithm requires only $\mathcal{O}(n \log^2 n)$ flops. In Refs. 11 and 12 we presented a fast algorithm for solving (block) Hankel systems. We first transformed the Hankel matrix into a Loewner matrix. Then we used an explicit formula for the inverse of a Loewner matrix. This formula involves certain parameters that can be computed by solving two linearized rational interpolation problems at roots of unity. We presented a fast algorithm for solving these interpolation problems and in this way, since the transformation from Hankel to Loewner and back to Hankel can be done via FFT in $\mathcal{O}(n \log n)$ flops, we obtained a fast Hankel solver. In Ref. 13 we assumed that the size of the Hankel matrix is a power of 2 and we replaced our fast interpolation algorithm by a divide-and-conquer approach. In this way we obtained a superfast Hankel solver. In Ref. 14 Toeplitz matrices of arbitrary size (not necessarily a power of 2) are considered and an inversion formula given by Heinig and Rost[15] is used to obtain an explicit formula for the inverse of a Toeplitz matrix. This inversion formula again involves certain parameters that can be calculated by solving two linearized rational interpolation problems at roots of unity. If N denotes the smallest power of 2 that is larger than or equal to the size of the Toeplitz matrix, then these interpolation problems involve $2N$ interpolation conditions. The data for the interpolation problems is obtained by evaluating the symbol of the Toeplitz matrix at the $2N$th roots of unity. This approach leads to a superfast Toeplitz solver.

By exploiting the connection between linearized rational interpolation problems and coupled Vandermonde matrices, we have recently shown how the determinant of a Hankel or a Toeplitz matrix can be computed in a superfast way from the "pivots" that appear in our interpolation algorithm.[16,17]

Further author information: (Send correspondence to Marc Van Barel)
Peter Kravanja: E-mail: Peter.Kravanja@na-net.ornl.gov
Marc Van Barel: E-mail: Marc.VanBarel@cs.kuleuven.ac.be

Classical superfast Toeplitz or Hankel solvers (e.g., the algorithms designed by Sugiyama et al.,[18] Bitmead and Anderson,[19] Brent, Gustavson and Yun,[20] Morf[21] and, more recently, by Musicus,[22] de Hoog[23] and Ammar and Gragg[24–26]) are only guaranteed to be numerically stable in case the matrix is Hermitian and positive definite. In fact, these algorithms are notoriously unstable when applied to indefinite matrices. Our approach gives good numerical results (even for indefinite systems of large size—our numerical experiments involve matrices of size up to $2^{18} = 262144$) because of the stabilizing techniques that we have developed: pivoting (we rearrange the order of the interpolation points), iterative refinement (based on an inversion formula for coupled Vandermonde matrices) and by giving what we have called "difficult" interpolation points an adequate treatment. For more details we refer to Refs. 13 and 14.

Our fast interpolation algorithm is based on the interpolation algorithm formulated by Van Barel and Bultheel,[27] which we used because pivoting, an important stabilizing technique, can be easily incorporated into it. It is a sequential algorithm in which a matrix polynomial that satisfies already some of the interpolation conditions is updated to satisfy two additional interpolation conditions. The updating matrix is a matrix polynomial of degree one. It is constructed in a two-step process that resembles Gaussian elimination. Recently, we have presented a new derivation of our updating formula,[16] which we will briefly recall in Sect. 3, after having introduced some notation and formulated the interpolation problem in Sect. 2. Then we will present two other approaches. In Sect. 4 we will define a discrete inner product based on roots of unity and we will introduce a new kind of updating matrix that is unitary with respect to this inner product. Linearized rational interpolation problems can also be seen as the limiting case of certain discrete least squares problems. Van Barel and Bultheel presented fast algorithms for solving such problems in case the least squares nodes belong to either the unit circle (not necessarily roots of unity) or the real axis.[28–31] In Sect. 5 we apply this approach to our interpolation problem.

Pivoting is not necessary in the least squares approach of Sect. 5 but it is an important stabilizing technique in the updating approach that resembles Gaussian elimination and in the approach that uses a unitary updating matrix. We will briefly discuss our pivoting strategies.

2. LINEARIZED RATIONAL INTERPOLATION

Let N be a positive integer and let the complex numbers s_k, e_k and f_k be given for $k = 1, \ldots, 2N$. Suppose that the s_k's are mutually distinct. Consider the following two linearized rational interpolation problems:

- Find polynomials $a(z), b(z) \in \mathbf{C}[z]$ such that
  - $a(z)$ is a monic polynomial of degree N,
  - $b(z)$ is of degree $< N$,
  - $e_k a(s_k) + f_k b(s_k) = 0$ for $k = 1, \ldots, 2N$.
- Find polynomials $c(z), d(z) \in \mathbf{C}[z]$ such that
  - $c(z)$ is of degree $< N$,
  - $d(z)$ is a monic polynomial of degree N,
  - $e_k c(s_k) + f_k d(s_k) = 0$ for $k = 1, \ldots, 2N$.

If we define the 2×2 matrix polynomial $B_N(z)$ as

$$B_N(z) := \begin{bmatrix} a(z) & c(z) \\ b(z) & d(z) \end{bmatrix} \in \mathbf{C}[z]^{2\times 2},$$

then we can combine both interpolation problems as follows: find a monic 2×2 matrix polynomial $B_N(z)$ of degree N such that

$$\begin{bmatrix} e_k & f_k \end{bmatrix} B_N(s_k) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $k = 1, \ldots, 2N$. Note that there are $2N$ interpolation conditions to be satisfied. We will assume that this interpolation problem has only one solution. (This holds for example if the interpolation problem is connected to a nonsingular Toeplitz matrix,[14] which is the case that interests us here. See also Ref. 32)

We will construct $B_N(z)$ as the last element in the sequence $B_0(z), B_1(z), \ldots, B_N(z)$, where $B_j(z) \in \mathbf{C}[z]^{2\times 2}$ is a monic matrix polynomial of degree j for $j = 0, 1, \ldots, N-1$. We define $B_0(z)$ as the 2×2 unit matrix,

$$B_0(z) := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

and for $j = 1, \ldots, N-1$ the matrix polynomial $B_j(z)$ is defined as the (again, assumed to be unique) monic 2×2 matrix polynomial of degree j that satisfies the interpolation conditions

$$\begin{bmatrix} e_k & f_k \end{bmatrix} B_j(s_k) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $k = 1, \ldots, 2j$.

We will use the following recurrence relation to compute the 2×2 matrix polynomials $B_1(z), \ldots, B_N(z)$:

$$B_{j+1}(z) = B_j(z) B_{j\to j+1}(z), \qquad j = 0, \ldots, N-1.$$

The 2×2 matrix polynomials $B_{j\to j+1}(z)$ will be determined such that

$$\deg B_{j\to j+1}(z) = 1 \qquad \text{and} \qquad \operatorname{hdc} B_{j\to j+1}(z) = I_2 \tag{1}$$

for $j = 0, \ldots, N-1$. Here 'hdc' denotes the highest degree coefficient and I_2 the 2×2 unit matrix. This already guarantees that $B_1(z), \ldots, B_N(z)$ satisfy the degree and monicity conditions. The degrees of freedom that remain in fixing $B_{j\to j+1}(z)$ are used to satisfy the interpolation conditions, i.e.,

$$\begin{bmatrix} e_k & f_k \end{bmatrix} B_j(s_k) B_{j\to j+1}(s_k) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $k = 1, \ldots, 2(j+1)$. Since

$$\begin{bmatrix} e_k & f_k \end{bmatrix} B_j(s_k) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $k = 1, \ldots, 2j$, we have to determine $B_{j\to j+1}(z)$ such that

$$\begin{bmatrix} e^\star & f^\star \end{bmatrix} B_j(s^\star) B_{j\to j+1}(s^\star) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

and

$$\begin{bmatrix} e^{\star\star} & f^{\star\star} \end{bmatrix} B_j(s^{\star\star}) B_{j\to j+1}(s^{\star\star}) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

where $s^\star := s_{2j+1}$, $s^{\star\star} := s_{2j+2}$ and similarly for $e^\star$, $e^{\star\star}$, $f^\star$ and $f^{\star\star}$. If we define

$$\begin{bmatrix} l^\star & r^\star \end{bmatrix} := \begin{bmatrix} e^\star & f^\star \end{bmatrix} B_j(s^\star)$$

and

$$\begin{bmatrix} l^{\star\star} & r^{\star\star} \end{bmatrix} := \begin{bmatrix} e^{\star\star} & f^{\star\star} \end{bmatrix} B_j(s^{\star\star}),$$

then $B_{j\to j+1}(z)$ has to satisfy

$$\begin{aligned} \begin{bmatrix} l^\star & r^\star \end{bmatrix} B_{j\to j+1}(s^\star) &= \begin{bmatrix} 0 & 0 \end{bmatrix}, \\ \begin{bmatrix} l^{\star\star} & r^{\star\star} \end{bmatrix} B_{j\to j+1}(s^{\star\star}) &= \begin{bmatrix} 0 & 0 \end{bmatrix}. \end{aligned} \tag{2}$$

We call $l^\star$ a *left residual*, $r^\star$ a *right residual*, and similarly for $l^{\star\star}$ and $r^{\star\star}$.

3. AN APPROACH THAT RESEMBLES GAUSSIAN ELIMINATION

Define the matrix polynomials $B^{(L)}_{j\to j+1}(z)$ and $B^{(R)}_{j\to j+1}(z)$ as

$$B^{(L)}_{j\to j+1}(z) := \begin{bmatrix} z - s_L & \alpha_L \\ 0 & 1 \end{bmatrix} \qquad \text{and} \qquad B^{(R)}_{j\to j+1}(z) := \begin{bmatrix} 1 & 0 \\ \alpha_R & z - s_R \end{bmatrix}$$

where $\alpha_L, \alpha_R, s_L, s_R \in \mathbf{C}$, $s_L \neq s_R$. We assume that the 2×2 matrix

$$\begin{bmatrix} l^\star & r^\star \\ l^{\star\star} & r^{\star\star} \end{bmatrix} \tag{3}$$

is nonsingular. Indeed, this can always be ensured by interchanging the order of the interpolation points. Thus it is impossible that both $l^\star$ and $r^\star$ are equal to zero. Suppose that $l^\star \neq 0$. Then let $s_L := s^\star$ and $\alpha_L := -r^\star/l^\star$. It follows that

$$\begin{bmatrix} l^\star & r^\star \end{bmatrix} B^{(L)}_{j\to j+1}(s^\star) = \begin{bmatrix} 0 & 0 \end{bmatrix}.$$

We call the construction of $B^{(L)}_{j\to j+1}(z)$ a "left step" towards $B_{j\to j+1}(z)$. We have that

$$\begin{bmatrix} l^{\star\star} & r^{\star\star} \end{bmatrix} B^{(L)}_{j\to j+1}(s^{\star\star}) = \begin{bmatrix} l^{\star\star}(s^{\star\star} - s^\star) & l^{\star\star}(-\frac{r^\star}{l^\star}) + r^{\star\star} \end{bmatrix} =: \begin{bmatrix} \tilde{l}^{\star\star} & \tilde{r}^{\star\star} \end{bmatrix}.$$

Since

$$l^\star \cdot \tilde{r}^{\star\star} = \det \begin{bmatrix} l^\star & r^\star \\ l^{\star\star} & r^{\star\star} \end{bmatrix}, \tag{4}$$

it holds that $\tilde{r}^{\star\star} \neq 0$. Now let $s_R := s^{\star\star}$ and $\alpha_R := -\tilde{l}^{\star\star}/\tilde{r}^{\star\star}$. Then

$$\begin{bmatrix} \tilde{l}^{\star\star} & \tilde{r}^{\star\star} \end{bmatrix} B^{(R)}_{j\to j+1}(s^{\star\star}) = \begin{bmatrix} 0 & 0 \end{bmatrix}.$$

The construction of $B^{(R)}_{j\to j+1}(z)$ is called a "right step" towards $B_{j\to j+1}(z)$. By combining both steps we obtain that

$$B^{(L)}_{j\to j+1}(z) B^{(R)}_{j\to j+1}(z) = \begin{bmatrix} z + (\alpha_L \alpha_R - s_L) & \alpha_L(z - s_R) \\ \alpha_R & z - s_R \end{bmatrix}. \tag{5}$$

This matrix polynomial is a candidate for $B_{j\to j+1}(z)$. It already satisfies the degree condition in (1) and the interpolation conditions (2). It does not satisfy the monicity condition yet. We obtain $B_{j\to j+1}(z)$ by multiplying the matrix polynomial in (5) to the right by the inverse of its highest degree coefficient:

$$B_{j\to j+1}(z) = B^{(L)}_{j\to j+1}(z) B^{(R)}_{j\to j+1}(z) \begin{bmatrix} 1 & \alpha_L \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} (z - s_L) + \alpha_L \alpha_R & \alpha_L(s_L - s_R - \alpha_L \alpha_R) \\ \alpha_R & (z - s_R) - \alpha_L \alpha_R \end{bmatrix}.$$

If $r^\star \neq 0$, then $B_{j\to j+1}(z)$ can be constructed in a similar way from a "right step" followed by a "left step."

These considerations lead to a fast algorithm for computing $B_N(z)$. A high-level formulation of the corresponding Levinson-type and Schur-type algorithms can be found in Ref. 16. If N is a power of 2, then a superfast algorithm can be obtained by using a divide-and-conquer approach.[13,14]

It goes without saying that a lot of technical details need to be closely examined before the theoretical results that we have briefly summarized so far can be turned into a numerically stable implementation. We have already mentioned that stabilizing techniques are necessary and enumerated the most important ones that we have introduced. More information can be found in Refs. 13 and 14. A preliminary Fortran 90 implementation of our algorithms is available.

Pivoting is an important stabilizing technique. We will now briefly discuss the pivoting strategy that we use in the updating strategy that resembles Gaussian elimination.

3.1. Pivoting Strategy

Our pivoting strategy is very simple. The algorithm looks for the residual that has the largest modulus. The corresponding interpolation point is taken as the next interpolation point. If this largest residual is a left residual, then the algorithm performs a left step. It then looks for the right residual that has the largest modulus, takes the corresponding interpolation point as the next interpolation point, and performs a right step. Otherwise, first a right step is performed and then the search for the largest residual is restricted to the left residuals.

4. AN APPROACH BASED ON UNITARY TRANSFORMATIONS

It should be clear by now that the updating matrix $B_{j\to j+1}(z)$ has played a key role in the development of our (super)fast algorithm for solving linearized rational interpolation problems and hence in our fast and superfast algorithms for solving (indefinite) Toeplitz or Hankel systems or for computing the determinant of a Toeplitz or a Hankel matrix.

We are now going to introduce a new updating matrix that can be used in case the interpolation points are roots of unity. Assume therefore from now on that the s_k's are the $2N$th roots of unity:

$$s_k := \exp\Big[\frac{2\pi i}{2N}(k-1)\Big], \qquad k = 1, \ldots, 2N.$$

Let $\mathcal{P}_{2N-1} \subset \mathbf{C}[z]$ denote the linear space of polynomials of degree $\leq 2N-1$. This degree bound ensures that the following linear and conjugate symmetric form defines a positive definite (discrete) inner product:

$$\langle\langle\, p(z), q(z)\,\rangle\rangle_{2N} := \frac{1}{2N}\sum_{k=1}^{2N} p(s_k)\,\overline{q(s_k)}$$

for each pair of polynomials $p(z), q(z) \in \mathcal{P}_{2N-1}$. Here the bar denotes the complex conjugate. The constant in front of the sum normalizes the inner product such that $\langle\langle 1, 1\rangle\rangle_{2N} = 1$. The inner product of two 2×1 column polynomial vectors is defined as follows:

$$\langle\langle \begin{bmatrix} p(z) \\ r(z) \end{bmatrix}, \begin{bmatrix} q(z) \\ s(z) \end{bmatrix} \rangle\rangle_{2N} := \langle\langle\, p(z), q(z)\,\rangle\rangle_{2N} + \langle\langle\, r(z), s(z)\,\rangle\rangle_{2N}$$

where $p(z), q(z), r(z), s(z) \in \mathcal{P}_{2N-1}$.

Let us for the time being forget about the monicity condition in (1) and search for a 2×2 matrix polynomial of degree one,

$$\mathcal{B}(z) := \begin{bmatrix} \alpha_1 z + \alpha_0 & \gamma_1 z + \gamma_0 \\ \beta_1 z + \beta_0 & \delta_1 z + \delta_0 \end{bmatrix} \in \mathbf{C}[z]^{2\times 2},$$

that satisfies the interpolation conditions (2),

$$\begin{bmatrix} l^\star & r^\star \end{bmatrix} \mathcal{B}(s^\star) = \begin{bmatrix} l^{\star\star} & r^{\star\star} \end{bmatrix} \mathcal{B}(s^{\star\star}) = \begin{bmatrix} 0 & 0 \end{bmatrix}. \tag{6}$$

The constants $\alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1, \delta_0, \delta_1 \in \mathbf{C}$ are to be determined. Equation (6) implies that

$$\begin{aligned} l^\star\alpha_0 + l^\star s^\star \alpha_1 + r^\star\beta_0 + r^\star s^\star \beta_1 &= 0 \\ l^\star\gamma_0 + l^\star s^\star \gamma_1 + r^\star\delta_0 + r^\star s^\star \delta_1 &= 0 \end{aligned}$$

and

$$\begin{aligned} l^{\star\star}\alpha_0 + l^{\star\star} s^{\star\star} \alpha_1 + r^{\star\star}\beta_0 + r^{\star\star} s^{\star\star} \beta_1 &= 0 \\ l^{\star\star}\gamma_0 + l^{\star\star} s^{\star\star} \gamma_1 + r^{\star\star}\delta_0 + r^{\star\star} s^{\star\star} \delta_1 &= 0. \end{aligned}$$

If we define

$$\vec{v} := \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \beta_0 \\ \beta_1 \end{bmatrix}, \quad \vec{w} := \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \delta_0 \\ \delta_1 \end{bmatrix}, \quad \vec{\lambda}^\star := \overline{\begin{bmatrix} l^\star \\ l^\star s^\star \\ r^\star \\ r^\star s^\star \end{bmatrix}} \quad \text{and} \quad \vec{\lambda}^{\star\star} := \overline{\begin{bmatrix} l^{\star\star} \\ l^{\star\star} s^{\star\star} \\ r^{\star\star} \\ r^{\star\star} s^{\star\star} \end{bmatrix}},$$

then these equations can be written as

$$\langle \vec{v}, \vec{\lambda}^\star \rangle = \langle \vec{v}, \vec{\lambda}^{\star\star} \rangle = 0 \qquad \text{and} \qquad \langle \vec{w}, \vec{\lambda}^\star \rangle = \langle \vec{w}, \vec{\lambda}^{\star\star} \rangle = 0. \tag{7}$$

Here the bar in the definition of $\vec{\lambda}^\star$ and $\vec{\lambda}^{\star\star}$ denotes the complex conjugate and $\langle \cdot, \cdot \rangle$ denotes the classical Euclidean inner product of complex column vectors. The associated norm will be denoted by $\|\cdot\|$. Note that the vectors $\vec{\lambda}^\star$ and

$\vec{\lambda}^{\star\star}$ are linearly independent as $s^\star$ is different from $s^{\star\star}$. The interpolation conditions (2) thus imply that the coefficient vectors $\vec{v}$ and $\vec{w}$ are orthogonal to the vectors $\vec{\lambda}^\star$ and $\vec{\lambda}^{\star\star}$. In other words, to determine a matrix polynomial $\mathcal{B}(z)$, from which we will derive the updating matrix $B_{j\to j+1}(z)$ by normalizing its highest degree coefficient, we need to find two vectors that belong to the orthogonal complement of the linear space spanned by $\vec{\lambda}^\star$ and $\vec{\lambda}^{\star\star}$.

The vectors $\vec{v}$ and $\vec{w}$ are of course not completely determined by the orthogonality conditions (7). In addition, we will demand that our matrix polynomial $\mathcal{B}(z)$ is a unitary matrix with respect to $\langle\langle\cdot,\cdot\rangle\rangle_{2N}$. This condition is expressed as

$$\langle\langle \begin{bmatrix} \alpha_1 z+\alpha_0 \\ \beta_1 z+\beta_0 \end{bmatrix}, \begin{bmatrix} \gamma_1 z+\gamma_0 \\ \delta_1 z+\delta_0 \end{bmatrix} \rangle\rangle_{2N} = 0 \tag{8}$$

and

$$\langle\langle \begin{bmatrix} \alpha_1 z+\alpha_0 \\ \beta_1 z+\beta_0 \end{bmatrix}, \begin{bmatrix} \alpha_1 z+\alpha_0 \\ \beta_1 z+\beta_0 \end{bmatrix} \rangle\rangle_{2N} = 1, \qquad \langle\langle \begin{bmatrix} \gamma_1 z+\gamma_0 \\ \delta_1 z+\delta_0 \end{bmatrix}, \begin{bmatrix} \gamma_1 z+\gamma_0 \\ \delta_1 z+\delta_0 \end{bmatrix} \rangle\rangle_{2N} = 1. \tag{9}$$

One can easily verify that Eqs. (8) and (9) are equivalent to

$$\overline{\alpha_0}\gamma_0 + \overline{\alpha_1}\gamma_1 + \overline{\beta_0}\delta_0 + \overline{\beta_1}\delta_1 = 0$$

and

$$\begin{aligned} |\alpha_0|^2 + |\alpha_1|^2 + |\beta_0|^2 + |\beta_1|^2 &= 1, \\ |\gamma_0|^2 + |\gamma_1|^2 + |\delta_0|^2 + |\delta_1|^2 &= 1. \end{aligned}$$

Thus the vectors $\vec{v}$ and $\vec{w}$ have to satisfy

$$\langle \vec{v}, \vec{w} \rangle = 0 \qquad \text{and} \qquad \|\vec{v}\| = \|\vec{w}\| = 1.$$

In other words, we are looking for an orthonormal basis for the orthogonal complement of the linear space spanned by $\vec{\lambda}^\star$ and $\vec{\lambda}^{\star\star}$.

One can easily verify that the highest degree coefficient of $\mathcal{B}(z)$ is nonsingular if the matrix (3) is nonsingular. Thus, it is always possible to obtain $B_{j\to j+1}(z)$ by normalizing $\mathcal{B}(z)$.

The vectors $\vec{v}$ and $\vec{w}$ can be obtained by computing the QR factorization of the 4×2 matrix

$$\begin{bmatrix} \vec{\lambda}^\star & \vec{\lambda}^{\star\star} \end{bmatrix}.$$

The last two columns of the unitary matrix $Q \in \mathbf{C}^{4\times 4}$ are candidates for $\vec{v}$ and $\vec{w}$.

4.1. Pivoting Strategy

The idea behind our pivoting strategy is to find vectors $\vec{\lambda}^\star$ and $\vec{\lambda}^{\star\star}$ that are already as orthogonal as possible while at the same time having a norm that is as large as possible. First we select the interpolation point $s^\star$ such that $\|\vec{\lambda}^\star\|$ is maximal. Then we choose $s^{\star\star}$ such that the norm of the component of $\vec{\lambda}^{\star\star}$ that is orthogonal to $\vec{\lambda}^\star$,

$$\|\vec{\lambda}^{\star\star} - \frac{\langle \vec{\lambda}^{\star\star}, \vec{\lambda}^\star \rangle}{\|\vec{\lambda}^\star\|^2}\vec{\lambda}^\star\|,$$

is maximal.

5. DISCRETE LEAST SQUARES RATIONAL APPROXIMATION

Consider the following discrete least squares problem. Determine polynomials $p(z), q(z) \in \mathbf{C}[z]$, $p(z)$ or $q(z)$ monic, such that

$$\sum_{k=1}^{2N} |e_k p(s_k) + f_k q(s_k)|^2$$

is minimal and the degree condition

$$\deg \begin{bmatrix} p(z) \\ q(z) \end{bmatrix} = N$$

is satisfied. Observe that both the vectors

$$\begin{bmatrix} a(z) \\ b(z) \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} c(z) \\ d(z) \end{bmatrix} \tag{10}$$

satisfy these conditions. In both cases the minimum is equal to zero. (The polynomial vectors in (10) are distinct as the degrees of the corresponding components are different.) In other words, our problem of computing the matrix polynomial $B_N(z)$ is transformed into that of solving a certain discrete least squares problem at the roots of unity s_k.

Van Barel and Bultheel presented fast algorithms for solving discrete least squares problems in case the least squares nodes belong to either the unit circle (not necessarily roots of unity) or the real axis.[28–31] On the unit circle, their approach is based on unitary similarity transformations that can be performed in a fast way by using the block Schur parametrization of a block unitary Hessenberg matrix. The algorithm computes the coefficients of the recurrence relation satisfied by the intermediary (i.e., with a lower degree) solutions to the least squares problem. The last two such solutions are the ones that we need. The coefficients (in the classical monomial basis) of these solutions can be computed in two ways. We can use the recurrence relation either to update the coefficients of the intermediary solutions, or to compute the values that the intermediary solutions take at the roots of unity and then obtain the coefficients of the last two solutions via polynomial interpolation (FFT). We have observed that the latter option leads to more accurate numerical results. Hence this is the option that we will use in the numerical examples.

6. NUMERICAL EXPERIMENTS

We consider 100 Toeplitz matrices T of size 256×256 whose entries are uniformly random distributed in $[0, 1]$. We define $x := [\ 1 \quad \cdots \quad 1\ ]^T \in \mathbf{C}^{256}$ and $b := Tx$. As described in Refs. 13 and 14, the Toeplitz system $Tx = b$ is connected to a linearized rational interpolation problem at roots of unity. We will now compare the results obtained by the three fast algorithms that we have presented.

Let $\hat{x}$ denote the computed solution to the Toeplitz system $Tx = b$. In Fig. 1 we plot a histogram for the accuracy obtained by our algorithms, as indicated by

$$-\log_{10} \frac{\|\hat{x} - x\|_2}{\|x\|_2}.$$

We consider the three approaches that we have discussed: Gaussian elimination (Sect. 3), updating matrices that are unitary with respect to a discrete inner product (Sect. 4) and discrete least squares (Sect. 5).

For this set of matrices the approach based on Gaussian elimination clearly emerges as the most accurate one. We have computed the coefficients of the polynomials that solve the linearized rational interpolation problem by updating the coefficients of the polynomials that solve the intermediary (i.e., with a smaller number of interpolation conditions) problems. However, cf. our discussion at the end of Sect. 5, one can also update the values taken by the intermediary polynomials at the roots of unity and then perform polynomial interpolation (FFT). The latter approach gives even more accurate results, as can be seen in Fig. 2.

In the previous experiment, which mainly involves well-conditioned matrices, the approach based on a unitary updating matrix gives the least satisfactory results. We have checked that this is also true for ill-conditioned matrices.

We now consider 100 possibly ill-conditioned banded Toeplitz matrices $T^\star$ of size 64×64. The matrices have seven non-zero bands (symmetric around the main diagonal) whose entries are uniformly random distributed in $[0, 1]$. The vectors $x^\star$ and $b^\star$ are defined in the same way as x and b above. In Figs. 3 and 4 we plot a histogram of the accuracy obtained by the approach based on Gaussian elimination and on discrete least squares. In both cases we have updated the values instead of the coefficients. In Fig. 3 we plot minus the logarithm with base 10 of the relative error of the computed solution to the Toeplitz system, and in Fig. 4 we plot minus the logarithm with base 10 of the norm of the residual vector divided by the norm of the right-hand side vector.

In Fig. 5 we plot a histogram of the logarithm with base 10 of the condition numbers (in the 2-norm) of the Toeplitz matrices $T^\star$.

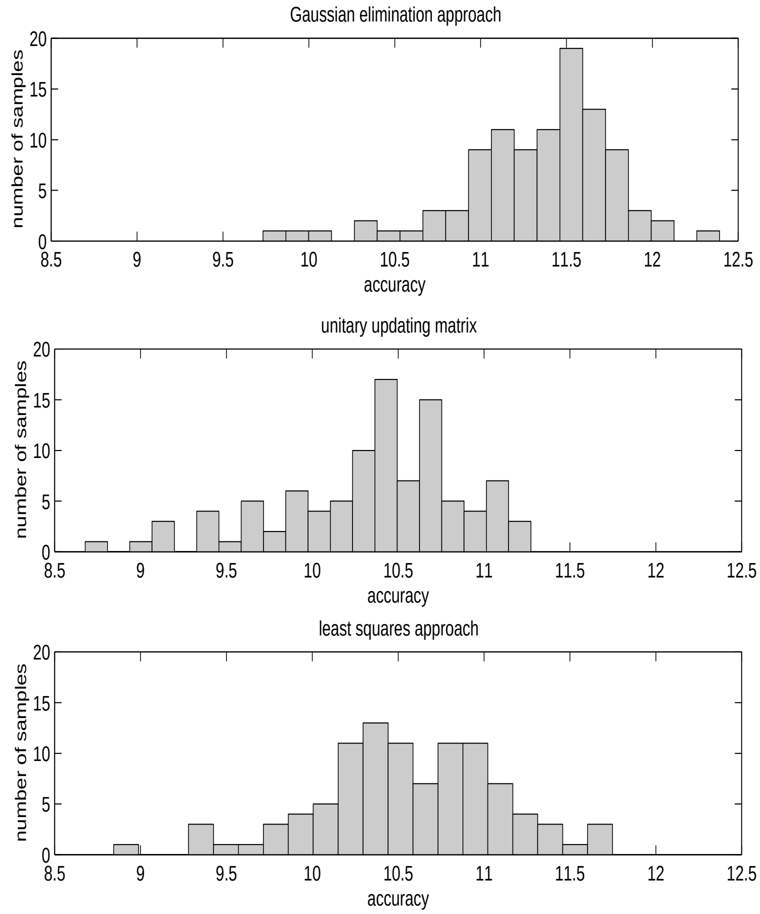

Figure 1. Histogram of the accuracy obtained by the three approaches.

7. CONCLUSION

We have compared three strategies for solving the linearized rational interpolation problems at roots of unity that are connected to the fast and superfast Toeplitz or Hankel solvers that we developed. Our numerical experiments indicate that the approach resembling Gaussian elimination gives the most accurate results. They also show that it is preferable to update the values of the intermediary polynomials instead of the coefficients.

ACKNOWLEDGMENTS

This research was partially supported by the Fund for Scientific Research-Flanders (FWO-V), project "Orthogonal Systems and Their Applications," grant #G.0278.97.

REFERENCES

1. J. Meinguet, "On the solubility of the Cauchy interpolation problem," in *Approximation Theory*, A. Talbot, ed., pp. 137–163, Academic Press, 1970.
2. A. C. Antoulas, "On the scalar rational interpolation problem," *IMA J. Math. Control Inf.* **3**, pp. 61–88, 1986.
3. A. C. Antoulas, "Rational interpolation and the Euclidean algorithm," *Linear Algebr. Appl.* **108**, pp. 157–171, 1988.
4. A. C. Antoulas, J. A. Ball, J. Kang, and J. C. Willems, "On the solution of the minimal rational interpolation problem," *Linear Algebr. Appl.* **137/138**, pp. 511–573, Aug./Sept. 1990.

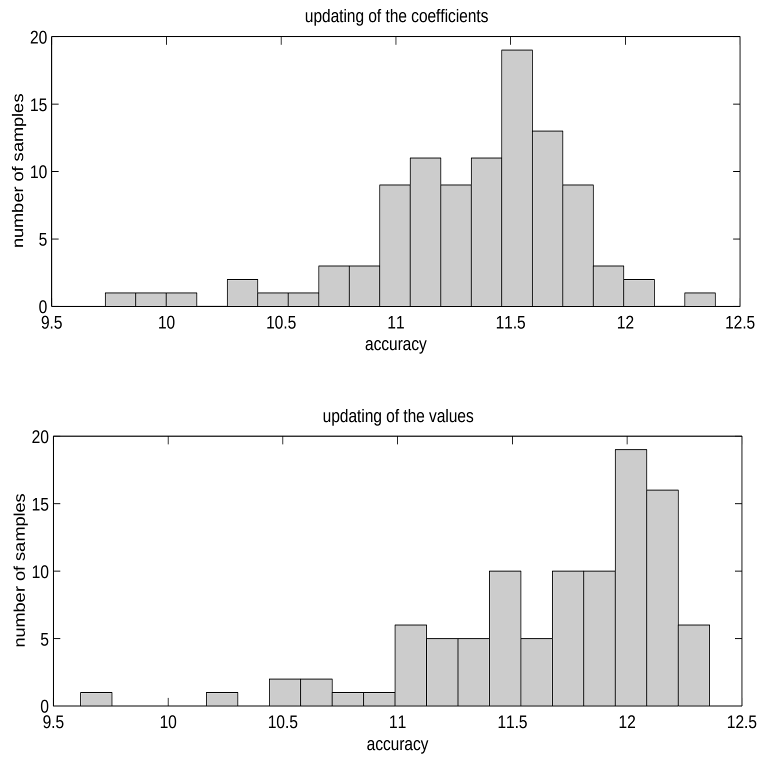

Figure 2. Histogram of the accuracy obtained by the two variants (coefficients versus values) of the Gaussian elimination approach.

5. J.-P. Berrut and H. D. Mittelmann, “Matrices for the direct determination of the barycentric weights of rational interpolation,” *J. Comput. Appl. Math.* **78**, pp. 355–370, Feb. 1997.
6. J.-P. Berrut and H. D. Mittelmann, “Rational interpolation through the optimal attachment of poles to the interpolating polynomial,” report No. 98–2, Université de Fribourg (Suisse), Institut de Mathématiques, June 1998.
7. M. H. Gutknecht, “Continued fractions associated with the Newton–Padé table,” *Numer. Math.* **56**, pp. 547–589, 1989.
8. M. H. Gutknecht, “In what sense is the rational interpolation problem well posed?,” *Constr. Approx.* **6**(4), pp. 437–450, 1990.
9. M. H. Gutknecht, “The rational interpolation problem revisited,” *Rocky Mt. J. Math.* **21**(1), pp. 263–280, 1991.
10. M. H. Gutknecht, “Block structure and recursiveness in rational interpolation,” in *Approximation Theory VII*, E. W. Cheney, C. K. Chui, and L. L. Schumaker, eds., pp. 93–130, Academic Press, 1992.
11. P. Kravanja and M. Van Barel, “A fast block Hankel solver based on an inversion formula for block Loewner matrices,” *CALCOLO* **33**, pp. 147–164, Jan.–June 1996. Proceedings of the workshop *Toeplitz Matrices: Structure, Algorithms and Applications*, Cortona (Italy), September 9–12, 1996. [Zbl 904.65031].
12. P. Kravanja and M. Van Barel, “A fast Hankel solver based on an inversion formula for Loewner matrices,” *Linear Algebr. Appl.* **282**(1–3), pp. 275–295, 1998.
13. M. Van Barel and P. Kravanja, “A stabilized superfast solver for indefinite Hankel systems,” *Linear Algebr. Appl.* **284**(1–3), pp. 335–355, 1998. Special issue on the International Linear Algebra Society Symposium “Linear Algebra in Control Theory, Signals and Image Processing,” held at the University of Manitoba, Canada, 6–8 June 1997.
14. P. Kravanja, *On Computing Zeros of Analytic Functions and Related Problems in Structured Numerical Linear Algebra.* PhD thesis, Katholieke Universiteit Leuven, Department of Computer Science, Mar. 1999.

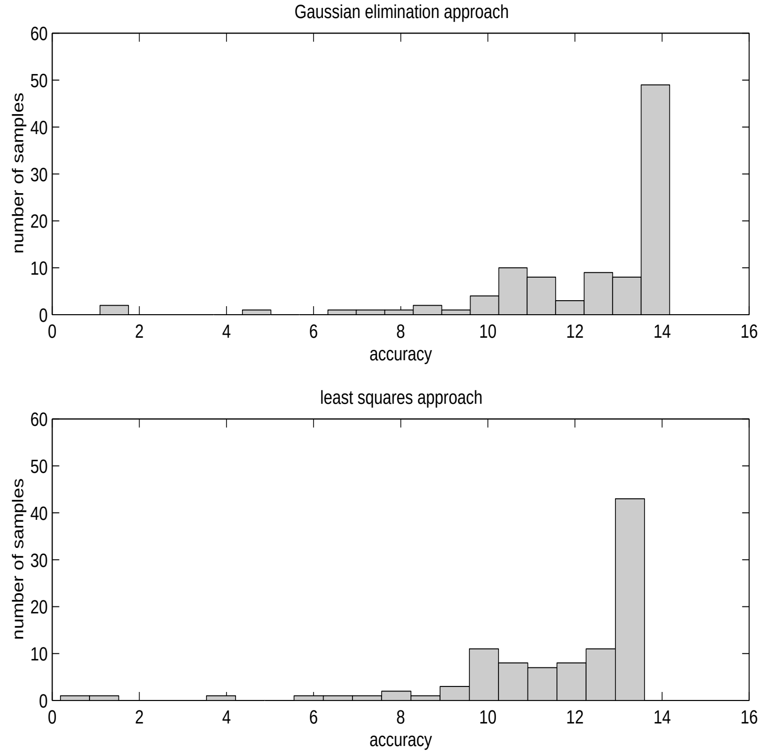

Figure 3. Histogram of the accuracy (relative error of the computed solution) obtained by the approaches based on Gaussian elimination and on discrete least squares.

15. G. Heinig and K. Rost, *Algebraic Methods for Toeplitz-like Matrices and Operators*, vol. 13 of *Operator Theory: Advances and Applications*, Birkhäuser, 1984.
16. P. Kravanja and M. Van Barel, "Coupled Vandermonde matrices and the superfast computation of Toeplitz determinants." Submitted to *Numerical Algorithms*. Proceedings of the *International Conference on Rational Approximation* (ICRA99), Antwerp (Belgium), June 6–12, 1999.
17. M. Van Barel and P. Kravanja, "On the generically superfast computation of Hankel determinants." To appear in the proceedings of the *Second Workshop on Large-Scale Scientific Computations*, Sozopol (Bulgaria), June 2–6, 1999.
18. Y. Sugiyama, M. Kasahara, S. Hirasawa, and T. Namekawa, "A new method for solving key equations for decoding Goppa codes," *Internat. J. Control* **27**, pp. 87–99, 1975.
19. R. R. Bitmead and B. D. O. Anderson, "Asymptotically fast solution of Toeplitz and related systems of linear equations," *Linear Algebr. Appl.* **34**, pp. 103–116, 1980.
20. R. Brent, F. Gustavson, and D. Yun, "Fast solution of Toeplitz systems of equations and computation of Padé approximants," *J. Algorithms* **1**, pp. 259–295, Sept. 1980.
21. M. Morf, "Doubling algorithms for Toeplitz and related equations," in *Proc. IEEE Internat. Conf. on Acoustics, Speech and Signal Processing*, pp. 954–959, (Denver, CO), 1980.
22. B. R. Musicus, "Levinson and fast Cholesky algorithms for Toeplitz and almost Toeplitz matrices," report, Res. Lab. of Electronics, M.I.T., 1984.
23. F. de Hoog, "A new algorithm for solving Toeplitz systems of equations," *Linear Algebr. Appl.* **88/89**, pp. 123–138, 1987.
24. G. S. Ammar and W. B. Gragg, "The generalized Schur algorithm for the superfast solution of Toeplitz systems," in *Rational Approximation and its Applications in Mathematics and Physics*, J. Gilewicz, M. Pindor, and W. Siemaszko, eds., vol. 1237 of *Lecture Notes in Mathematics*, pp. 315–330, Springer, 1987.

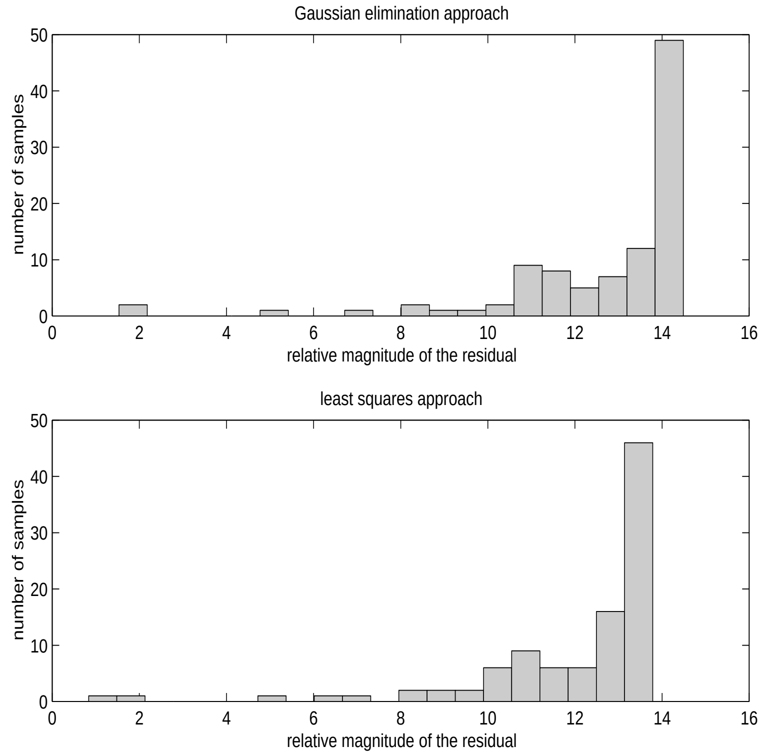

Figure 4. Histogram of the accuracy (relative magnitude of the residual vector) obtained by the approaches based on Gaussian elimination and on discrete least squares.

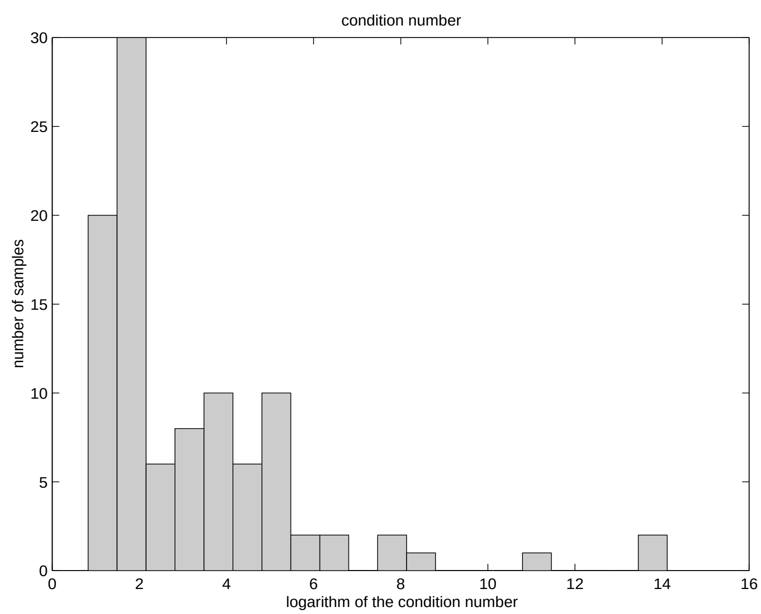

Figure 5. Histogram of the condition numbers of the Toeplitz matrices $T^{\star}$.

25. G. S. Ammar and W. B. Gragg, "Superfast solution of real positive definite Toeplitz systems," *SIAM J. Matrix Anal. Appl.* **9**, pp. 61–76, Jan. 1988.
26. G. S. Ammar and W. B. Gragg, "Numerical experience with a superfast real Toeplitz solver," *Linear Algebr. Appl.* **121**, pp. 185–206, 1989.
27. M. Van Barel and A. Bultheel, "A new approach to the rational interpolation problem," *J. Comput. Appl. Math.* **32**, pp. 281–289, 1990.
28. M. Van Barel and A. Bultheel, "A parallel algorithm for discrete least squares rational approximation," *Numer. Math.* **63**, pp. 99–121, 1992.
29. M. Van Barel and A. Bultheel, "Discrete linearized least squares approximation on the unit circle," *J. Comput. Appl. Math.* **50**, pp. 545–563, 1994.
30. M. V. Barel and A. Bultheel, "Orthonormal polynomial vectors and least squares approximation for a discrete inner product," *Electronic Transactions on Numerical Analysis* **3**, pp. 1–23, March 1995.
31. A. Bultheel and M. Van Barel, "Vector orthogonal polynomials and least squares approximation," *SIAM J. Matrix Anal. Appl.* **16**(3), pp. 863–885, 1995.
32. M. Van Barel and A. Bultheel, "A general module theoretic framework for vector M-Padé and matrix rational interpolation," *Numer. Algorithms* **3**, pp. 451–462, 1992.