On the generically superfast computation of Hankel determinants

Marc Van Barel and Peter Kravanja
Katholieke Universiteit Leuven, Department of Computer Science
Celestijnenlaan 200 A, B-3001 Heverlee (Belgium)

E-mail: `Marc.VanBarel@cs.kuleuven.ac.be`,
`Peter.Kravanja@na-net.ornl.gov`

Summary

We present a generically superfast algorithm for computing the determinant of a complex Hankel matrix whose size is a power of 2. Our approach exploits the connections that exist between Hankel, Loewner, Cauchy and (coupled) Vandermonde matrices. We show that the determinant of a Hankel matrix can be computed from the determinant of a certain coupled Vandermonde matrix. The latter matrix is related to a linearized rational interpolation problem at roots of unity and we show how its determinant can be calculated by multiplying the pivots that appear in the generically superfast interpolation algorithm that we presented in a previous paper.

1 Introduction

Let n be a power of 2 and let $H = H_n := [h_{k+l}]_{k,l=0}^{n-1}$ be a nonsingular $n \times n$ complex Hankel matrix. We consider the problem of computing the determinant of H. By exploiting the connections between Hankel, Loewner, Cauchy, Vandermonde and coupled Vandermonde matrices, we will show how the determinant of H can be computed from the determinant of a coupled Vandermonde matrix.

In our paper [5] we presented a generically superfast algorithm for linearized rational interpolation at roots of unity. Superfast means that the arithmetic complexity of our algorithm is $\mathcal{O}(N \log^2 N)$ for a problem that consists of N interpolation points. Generically refers to the fact that in some exceptional cases the complexity of the algorithm is only $\mathcal{O}(N^2)$. Linearized rational interpolation problems can be written in terms of coupled Vandermonde matrices. We will show how the determinant of such a matrix can be computed from the pivots that appear in our interpolation algorithm. This will enable us to compute $\det H$ in a generically superfast way.

The restriction that n has to be a power of 2 comes from the fact that our interpolation algorithm can only be applied in case N is of the form $N = 2^{p+1}$ for some $p \in \mathbb{N}$.

2 Loewner, Cauchy and (coupled) Vandermonde matrices

Let $y_1, \dots, y_n, z_1, \dots, z_n$ be $2n$ mutually distinct complex numbers and define $\mathbf{y} := (y_1, \dots, y_n)$ and $\mathbf{z} := (z_1, \dots, z_n)$. Let $\mathcal{L}(\mathbf{y},\mathbf{z})$ be the class of matrices

$$\mathcal{L}(\mathbf{y},\mathbf{z}) := \left\{ \left[\frac{c_k - d_l}{y_k - z_l} \right]_{k,l=1}^{n} \mid c_1, \dots, c_n, d_1, \dots, d_n \in \mathbb{C} \right\}.$$

The elements of $\mathcal{L}(\mathbf{y},\mathbf{z})$ are called *Loewner matrices*. They bear the name of Karl Loewner who studied them in the context of rational interpolation and monotone matrix functions [4].

The set $\mathcal{L}(\mathbf{y},\mathbf{z})$ is a linear space over $\mathbb{C}$ and a subspace of the linear space of all the $n \times n$ complex matrices. Since addition of a constant to all the $2n$ parameters c_k, d_l leads to the same Loewner matrix, its dimension is $2n-1$. The set of all the $n \times n$ complex Hankel matrices also forms a linear subspace of dimension $2n-1$. Hankel and Loewner matrices are even more closely related. According to Fiedler [1] every Hankel matrix can be transformed into a Loewner matrix and vice versa.

Before we can formulate this theorem, we first have to deal with some preliminaries concerning Vandermonde matrices. Let $t_1, \dots, t_n$ be n complex numbers and define $\mathbf{t} := (t_1, \dots, t_n)$. The *Vandermonde matrix* with nodes $t_1, \dots, t_n$ is given by $V(\mathbf{t}) = V(t_1, \dots, t_n) := [t_k^{l-1}]_{k,l=1}^{n}$. Let $f_{\mathbf{t}}(z)$ be the monic polynomial of degree n that has zeros $t_1, \dots, t_n$, $f_{\mathbf{t}}(z) := (z - t_1) \cdots (z - t_n)$, and define $f_{\mathbf{t},j}(z) := \prod_{k \neq j} (z - t_k)$ for $j = 1, \dots, n$. Note that $f_{\mathbf{t},j}(z)$ is a monic polynomial of degree $n-1$ for $j = 1, \dots, n$. Define the $n \times n$ matrix $W(\mathbf{t})$ by the equation

$$\begin{bmatrix} f_{\mathbf{t},1}(z) \\ f_{\mathbf{t},2}(z) \\ \vdots \\ f_{\mathbf{t},n}(z) \end{bmatrix} = W(\mathbf{t}) \begin{bmatrix} 1 \\ z \\ \vdots \\ z^{n-1} \end{bmatrix}. \tag{2.1}$$

This means that the jth row of $W(\mathbf{t})$ contains the coefficients of $f_{\mathbf{t},j}(z)$ when written in terms of the standard monomial basis $\{1, z, \dots, z^{n-1}\}$. Then

$$W(\mathbf{t})\,[V(\mathbf{t})]^T = \operatorname{diag}\left(f_{\mathbf{t},1}(t_1), \dots, f_{\mathbf{t},n}(t_n) \right). \tag{2.2}$$

The Vandermonde matrix $V(\mathbf{t})$ is nonsingular if and only if its nodes $t_1, \dots, t_n$ are mutually distinct. In that case (2.2) implies that $W(\mathbf{t})$ is nonsingular.

Let $V(\mathbf{y},\mathbf{z})$ be the $2n \times 2n$ Vandermonde matrix with nodes $y_1, \dots, y_n$ and $z_1, \dots, z_n$ and similarly for $W(\mathbf{y},\mathbf{z})$.

Theorem 1 *The matrix $L := W(\mathbf{y})\, H\, [W(\mathbf{z})]^T$ is a Loewner matrix in $\mathcal{L}(\mathbf{y},\mathbf{z})$ whose parameters $c_1,\ldots,c_n,d_1,\ldots,d_n$ are given by (up to an arbitrary additive constant $\xi \in \mathbb{C}$)*

$$\begin{bmatrix} c_1 \\ \vdots \\ c_n \\ d_1 \\ \vdots \\ d_n \end{bmatrix} = W(\mathbf{y},\mathbf{z}) \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{2n-2} \\ \xi \end{bmatrix}.$$

*S*ee Fiedler [1], Theorem 12. $\square$

Note that L is nonsingular.

A judicious choice of the points $\mathbf{y}$ and $\mathbf{z}$ enables us to write the transformation from H to L in terms of unitary matrices. Let $\omega := \exp(2\pi\, i/n)$ and suppose from now on that $y_k = \omega^{k-1}$ for $k = 1,\ldots,n$. That is, let $\mathbf{y} = (1,\omega,\ldots,\omega^{n-1})$. Let $\zeta := \exp(\pi\, i/n)$ and suppose from now on that $z_k = \zeta y_k$ for $k = 1,\ldots,n$. That is, let $\mathbf{z} = (\zeta,\zeta\omega,\ldots,\zeta\omega^{n-1})$. Let Ω_n be the $n \times n$ Fourier matrix,

$$\Omega_n := \frac{1}{\sqrt{n}} V(1,\omega,\ldots,\omega^{n-1}). \tag{2.3}$$

Matrix-vector products involving Ω_n (Ω_n^H) amount to a(n) (inverse) discrete Fourier transform (DFT) and can be evaluated via the celebrated (inverse) fast Fourier transform (FFT) in $\mathcal{O}(n \log n)$ flops. Finally, let $D_{n,\omega}$ and $D_{n,\zeta}$ be the $n \times n$ diagonal matrices $D_{n,\omega} := \mathrm{diag}\,(1,\omega,\ldots,\omega^{n-1})$ and $D_{n,\zeta} := \mathrm{diag}\,(1,\zeta,\ldots,\zeta^{n-1})$.

Proposition 2 *The matrices $W(\mathbf{y})$ and $[W(\mathbf{z})]^T$ can be written as*

$$W(\mathbf{y}) = \sqrt{n}\, \overline{D_{n,\omega}\Omega_n} \qquad \text{and} \qquad [W(\mathbf{z})]^T = \sqrt{n}\, \zeta^{n-1} \overline{D_{n,\zeta}}\, \Omega_n^H\, \overline{D_{n,\omega}}.$$

*S*ee Kravanja and Van Barel [3]. $\square$

These formulae imply that $W(\mathbf{y})/\sqrt{n}$ and $[W(\mathbf{z})]^T/\sqrt{n}$ are unitary. It follows that $\det L = n^n \alpha \det H$ where the modulus of $\alpha \in \mathbb{C}$ is equal to one.

The Loewner matrix L can be written as

$$L = \left[\frac{c_k - d_l}{y_k - z_l} \right]_{k,l=1}^{n} = \left[\frac{c_k}{y_k - z_l} \right]_{k,l=1}^{n} - \left[\frac{d_l}{y_k - z_l} \right]_{k,l=1}^{n} = D(\mathbf{c})C - CD(\mathbf{d})$$

where $D(\mathbf{c}) := \mathrm{diag}\,(c_1,\ldots,c_n)$, $D(\mathbf{d}) := \mathrm{diag}\,(d_1,\ldots,d_n)$ and C is the Cauchy matrix

$$C := \left[\frac{1}{y_k - z_l} \right]_{k,l=1}^{n}.$$

We will now derive a representation of the Cauchy matrix C that involves the Vandermonde matrices $V(\mathbf{y})$ and $V(\mathbf{z})$.

Proposition 3 $C = \left[\mathrm{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^{n}\right]^{-1} V(\mathbf{y})[V(\mathbf{z})]^{-1}\, \mathrm{diag}\left(f_{\mathbf{z},l}(z_l)\right)_{l=1}^{n}.$

Let $\alpha = [\alpha_k]_{k=1}^n$ and $\beta = [\beta_k]_{k=1}^n$ be two vectors in $\mathbb{C}^{n\times 1}$. Consider the system of linear equations $C\alpha = \beta$. This can be written as

$$\begin{aligned} & \sum_{l=1}^n \frac{1}{y_k - z_l}\, \alpha_l = \beta_k, \qquad k = 1, \ldots, n \\ \Leftrightarrow\quad & \sum_{l=1}^n \alpha_l\, \frac{f_{\mathbf{z},l}(y_k)}{f_{\mathbf{z}}(y_k)} = \beta_k, \qquad k = 1, \ldots, n \\ \Leftrightarrow\quad & \frac{p(y_k)}{f_{\mathbf{z}}(y_k)} = \beta_k, \qquad k = 1, \ldots, n \end{aligned}$$

where the polynomial $p(t)$ is defined as $p(t) := \sum_{l=1}^n \alpha_l\, f_{\mathbf{z},l}(t)$. Note that $\deg p(t) \leq n-1$. The vector α contains the coefficients of $p(t)$ with respect to the basis $\{f_{\mathbf{z},l}(t)\}_{l=1}^n$. If we write $p(t)$ in terms of the standard monomial basis $\{t^{l-1}\}_{l=1}^n$, $p(t) =: \sum_{l=1}^n \tilde{\alpha}_l\, t^{l-1}$, then the connection between the vectors α and $\tilde{\alpha} := [\tilde{\alpha}_l]_{l=1}^n$ is given by

$$\alpha^T\, W(\mathbf{z}) = \tilde{\alpha}^T. \tag{2.4}$$

This follows immediately from Equation (2.1). The fact that $p(y_k) = \beta_k f_{\mathbf{z}}(y_k)$ for $k = 1, \ldots, n$ immediately implies that

$$\begin{bmatrix} p(y_1) \\ \vdots \\ p(y_n) \end{bmatrix} = V(\mathbf{y})\,\tilde{\alpha} = \operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n \beta. \tag{2.5}$$

Equation (2.4) implies that $[W(\mathbf{z})]^T\alpha = \tilde{\alpha}$. Equation (2.5) then implies that $V(\mathbf{y})[W(\mathbf{z})]^T\alpha = \operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n \beta$, or, as $C\alpha = \beta$, that $V(\mathbf{y})[W(\mathbf{z})]^T C^{-1}\beta = \operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n \beta$. As β is arbitrary, it follows that

$$V(\mathbf{y})[W(\mathbf{z})]^T C^{-1} = \operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n. \tag{2.6}$$

Equation (2.2) implies that $V(\mathbf{z})[W(\mathbf{z})]^T = \operatorname{diag}\left(f_{\mathbf{z},l}(z_l)\right)_{l=1}^n$ and hence

$$[W(\mathbf{z})]^T = [V(\mathbf{z})]^{-1} \operatorname{diag}\left(f_{\mathbf{z},l}(z_l)\right)_{l=1}^n. \tag{2.7}$$

By combining (2.6) and (2.7) one easily obtains that indeed

$$C = \left[\operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n\right]^{-1} V(\mathbf{y})[V(\mathbf{z})]^{-1} \operatorname{diag}\left(f_{\mathbf{z},l}(z_l)\right)_{l=1}^n.$$

This proves the proposition. $\square$

We define the *coupled Vandermonde matrix* V_C as

$$V_C := \begin{bmatrix} V(\mathbf{y}) & -D(\mathbf{c})V(\mathbf{y}) \\ V(\mathbf{z}) & -D(\mathbf{d})V(\mathbf{z}) \end{bmatrix} \in \mathbb{C}^{2n\times 2n}.$$

The following theorem shows that $\det H$ can be easily computed from $\det V_C$.

Theorem 4 $\det V_C = i\left[\frac{2n}{i}\right]^n \det H$.

*T*he Schur complement formula implies that

$$\begin{aligned}
\det V_C &= \det V(\mathbf{y}) \det\left[-D(\mathbf{d})V(\mathbf{z}) + V(\mathbf{z})[V(\mathbf{y})]^{-1}D(\mathbf{c})V(\mathbf{y})\right] \\
&= \det V(\mathbf{y}) \det V(\mathbf{z}) \det\left[[V(\mathbf{y})]^{-1}D(\mathbf{c})V(\mathbf{y}) - [V(\mathbf{z})]^{-1}D(\mathbf{d})V(\mathbf{z})\right] \\
&= [\det V(\mathbf{z})]^2 \det\left[D(\mathbf{c})V(\mathbf{y})[V(\mathbf{z})]^{-1} - V(\mathbf{y})[V(\mathbf{z})]^{-1}D(\mathbf{d})\right].
\end{aligned}$$

Now let $D_1 := \operatorname{diag}\left(f_{\mathbf{z}}(y_k)\right)_{k=1}^n$ and $D_2 := \operatorname{diag}\left(f_{\mathbf{z},l}(z_l)\right)_{l=1}^n$ be the matrices that appear in Proposition 3. Then

$$\begin{aligned}
\det V_C &= \frac{\det D_1}{\det D_2}[\det V(\mathbf{z})]^2 \det\left[D(\mathbf{c})C - CD(\mathbf{d})\right] \\
&= \frac{\det D_1}{\det D_2}[\det V(\mathbf{z})]^2 \det L.
\end{aligned}$$

One can easily verify that $f_{\mathbf{z}}(t) = t^n - \zeta^n = t^n + 1$. Hence $f_{\mathbf{z}}(y_k) = 2$ for $k = 1,\ldots,n$ and thus $\det D_1 = 2^n$. Also, Equation (2.2) implies that $D_2 = W(\mathbf{z})[V(\mathbf{z})]^T$. It follows that

$$\det V_C = 2^n \frac{\det V(\mathbf{z})}{\det[W(\mathbf{z})]^T} \det L = 2^n \det V(\mathbf{z}) \det W(\mathbf{y}) \det H,$$

where we have used Theorem 1. Since $V(\mathbf{z}) = V(\mathbf{y})D_{n,\zeta}$, $\Omega_n = V(\mathbf{y})/\sqrt{n}$ and $W(\mathbf{y}) = \sqrt{n}\,\overline{D_{n,\omega}\Omega_n}$ (Proposition 2), it follows that

$$\begin{aligned}
\det V_C &= 2^n |\det V(\mathbf{y})|^2 \det D_{n,\zeta}\, \overline{\det D_{n,\omega}} \det H \\
&= (2n)^n \det D_{n,\zeta}\, \overline{\det D_{n,\omega}} \det H.
\end{aligned}$$

One can easily verify that $\det D_{n,\zeta} = \zeta^{\frac{(n-1)n}{2}}$ and $\det D_{n,\omega} = \omega^{\frac{(n-1)n}{2}}$. Hence

$$\begin{aligned}
\det D_{n,\zeta}\, \overline{\det D_{n,\omega}} &= \exp\left[\frac{(n-1)n}{2}\frac{\pi i}{n} - \frac{(n-1)n}{2}\frac{2\pi i}{n}\right] \\
&= \exp\left[-\frac{(n-1)n}{2}\frac{\pi i}{n}\right] = \left[\exp\left(\frac{\pi i}{2}\right)\right]^{-(n-1)} = i^{-(n-1)}.
\end{aligned}$$

It follows that $\det V_C = i\left[\frac{2n}{i}\right]^n \det H$. This proves the theorem. $\square$

3 Superfast rational interpolation

Let $a(z)$ and $b(z)$ be polynomials that satisfy the following degree conditions: $a(z)$ is a monic polynomial of degree n and $b(z)$ is a polynomial of degree $< n$.

Suppose also that the following linearized rational interpolation conditions are satisfied: $a(y_k) - c_k b(y_k) = 0$ and $a(z_l) - d_l b(z_l) = 0$ for $k,l = 1,\ldots,n$. One can prove that these polynomials are uniquely determined by these conditions [6]. Similarly, let $c(z)$ be a polynomial of degree $< n$ and $d(z)$ a monic polynomial of degree n such that $c(y_k) - c_k d(y_k) = 0$ and $c(z_l) - d_l d(z_l) = 0$ for $k,l = 1,\ldots,n$. Again, these polynomials are uniquely determined. Let us combine both interpolation problems by considering the matrix polynomial

$$B^\star(z) := \begin{bmatrix} a(z) & c(z) \\ b(z) & d(z) \end{bmatrix} \in \mathbb{C}[z]^{2\times 2}.$$

Then $B^\star(z)$ is the only monic 2×2 matrix polynomial that satisfies

$$\begin{bmatrix} 1 & -c_k \end{bmatrix} B^\star(y_k) = \begin{bmatrix} 0 & 0 \end{bmatrix} \qquad \text{and} \qquad \begin{bmatrix} 1 & -d_l \end{bmatrix} B^\star(z_l) = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $k,l = 1,\ldots,n$.

In our paper [3] we presented a sequential algorithm for computing $B^\star(z)$. The algorithm starts with the initialization $B_0(z) := I_2$ and then constructs the sequence $B_1(z),\ldots,B_n(z)$ where $B_k(z)$ is a 2×2 matrix polynomial of degree k for $k = 1,\ldots,n$ and $B_n(z) \equiv B^\star(z)$. After each step two additional interpolation conditions are satisfied. In fact, each step can be seen as a combination of two substeps, each of which handles one additional interpolation condition. Also, there are two different types of substeps: a "left" substep and a "right" substep and a step can consist of either the combination left+right or right+left. More details can be found in [3]. Our algorithm is a fast algorithm: it has arithmetic complexity $\mathcal{O}(N^2)$ where $N := 2n$. In [5] we used a divide-and-conquer approach to obtain a superfast $\mathcal{O}(N \log^2 N)$ version of the algorithm.

Let us define the matrix $\tilde{V}_C \in \mathbb{C}^{2n\times 2n}$ as follows. For $k = 1,\ldots,n$ the kth row of $\tilde{V}_C$ is given by

$$\left[\begin{array}{cc|cc|c|cc} 1 & -c_k & y_k & -c_k y_k & \cdots & y_k^{n-1} & -c_k y_k^{n-1} \end{array}\right]$$

and for $l = 1,\ldots,n$ the $(n+l)$th row of $\tilde{V}_C$ is given by

$$\left[\begin{array}{cc|cc|c|cc} 1 & -d_l & z_l & -d_l z_l & \cdots & z_l^{n-1} & -d_l z_l^{n-1} \end{array}\right].$$

Note that $\tilde{V}_C$ is obtained by reordering the columns of V_C. One can show that $\det \tilde{V}_C = (-1)^\alpha \det V_C$ where $\alpha = \sum_{j=1}^{n-1} j = (n-1)n/2$. The determinant of $\tilde{V}_C$, and hence also $\det V_C$, can be easily computed from information generated by our interpolation algorithm. Indeed, if $B(z)$ is any 2×2 matrix polynomial, then the *residual* is defined as the matrix in $\mathbb{C}^{2n\times 2}$ whose kth row is given by $\begin{bmatrix} 1 & -c_k \end{bmatrix} B(y_k)$ for $k = 1,\ldots,n$ and whose $(n+l)$th row is given by $\begin{bmatrix} 1 & -d_l \end{bmatrix} B(z_l)$ for $l = 1,\ldots,n$. If $B(z) \equiv B^\star(z)$, then all the entries of the residual are equal to zero. One can easily verify that the residual is given by $\tilde{V}_C[\mathcal{B}_k]_{k=0}^{n-1}$ in case $\deg B(z) < n$ and $B(z) =: \sum_{j=0}^{n-1} \mathcal{B}_j z^j$ where $\mathcal{B}_j \in \mathbb{C}^{2\times 2}$ for $j = 0,1,\ldots,n-1$. The block vector $[\mathcal{B}_k]_{k=0}^{n-1}$ is called the *stacking vector* of $B(z)$.

This leads to the following important observation. Let $\hat{B}_0,\hat{B}_1,\ldots,\hat{B}_{n-1}$ be the stacking vectors of the matrix polynomials $B_0(z),B_1(z),\ldots,B_{n-1}(z)$ (considered as polynomials of degree $\leq n-1$) generated by our interpolation algorithm. Then $\tilde{V}_C[\begin{array}{cccc} \hat{B}_0 & \hat{B}_1 & \cdots & \hat{B}_{n-1} \end{array}] =: R$ is a block lower triangular matrix whose block main diagonal consists of 2×2 matrices. The zero entries in R correspond to interpolation conditions that are satisfied. The more our algorithm proceeds, the more zeros appear. Note that $[\begin{array}{cccc} \hat{B}_0 & \hat{B}_1 & \cdots & \hat{B}_{n-1} \end{array}]$ is a block upper triangular matrix as $\deg B_k(z) = k$ for $k = 0,1,\ldots,n$. We are going to show that

$$\det[\begin{array}{cccc} \hat{B}_0 & \hat{B}_1 & \cdots & \hat{B}_{n-1} \end{array}] = 1. \tag{3.8}$$

This implies that $\det \tilde{V}_C$, and hence $(-1)^{\frac{(n-1)n}{2}} \det V_C$, is equal to the product of the determinants of the 2×2 blocks on the block main diagonal of R. We will investigate these determinants in a moment. Suppose that our interpolation algorithm takes a left+right step to go from $B_{k-1}(z)$ to $B_k(z)$ where $k \in \{1,\ldots,n\}$. Then

$$B_k(z) \equiv B_{k-1}(z) \left[\begin{array}{cc} z - s_L & \alpha_L \\ 0 & 1 \end{array}\right] \left[\begin{array}{cc} 1 & 0 \\ \alpha_R & z - s_R \end{array}\right]$$

for certain $s_L,s_R,\alpha_L,\alpha_R \in \mathbb{C}$ (for more details, we refer to [5]). It follows that

$$\text{hdc } B_k(z) = \text{hdc } B_{k-1}(z) \left[\begin{array}{cc} 1 & \alpha_L \\ 0 & 1 \end{array}\right]$$

where hdc denotes the highest degree coefficient. A similar result holds in case the algorithm takes a right+left step. Hence $\det \text{hdc } B_k(z) = \det \text{hdc } B_{k-1}(z)$ for $k = 1,\ldots,n$. As $B_0(z) := I_2$, it follows that $\det \text{hdc } B_k(z) = 1$ for $k = 0,1,\ldots,n$. This proves (3.8).

Let us now consider the 2×2 matrices that appear on the block main diagonal of R. Consider the kth block column of R where $k \in \{1,\ldots,n\}$. This block column contains the residual that corresponds to $B_{k-1}(z)$. Let D denote the 2×2 submatrix of this block column that is located on the block main diagonal of R. Suppose that our algorithm takes a left+right step to go from $B_{k-1}(z)$ to $B_k(z)$. In the left substep the (1,1)-entry in D is chosen as pivot and D is modified such that its (1,2)-entry becomes zero. The (2,2)-entry in the resulting matrix is chosen as pivot in the right substep. This can be summarized as follows:

$$\left[\begin{array}{cc} \pi_L & \star \\ \star & \star \end{array}\right] \left[\begin{array}{cc} 1 & \alpha_L \\ 0 & 1 \end{array}\right] = \left[\begin{array}{cc} \pi_L & 0 \\ \star & \pi_R \end{array}\right]$$

where $\pi_L,\alpha_L,\pi_R \in \mathbb{C}$. It follows that $\det D = \pi_L \pi_R$, in other words: the determinant of the 2×2 diagonal block is equal to the product of the pivot that is used in the left substep and the pivot that is used in the right substep. Now suppose that the algorithm takes a right+left step to go from $B_{k-1}(z)$ to $B_k(z)$. In this case the following holds:

$$\left[\begin{array}{cc} \star & \pi_R \\ \star & \star \end{array}\right] \left[\begin{array}{cc} 1 & 0 \\ \alpha_R & 1 \end{array}\right] = \left[\begin{array}{cc} 0 & \pi_R \\ \pi_L & \star \end{array}\right]$$

where π_L,α_R,$\pi_R \in \mathbb{C}$. Therefore $\det D = -\pi_L \pi_R$, in other words: the determinant of the 2×2 diagonal block is equal to minus the product of the pivot that is used in the right substep and the pivot that is used in the left substep. These observations lead to the following important conclusion: $\det \tilde{V}_C$ is equal to α times the product of the pivots that appear in our interpolation algorithm, where $\alpha = \alpha_1 \cdots \alpha_n$ and $\alpha_k \in \{1, -1\}$ for $k = 1, \ldots, n$. If the algorithm takes a left+right step to go from $B_{k-1}(z)$ to $B_k(z)$, then $\alpha_k = 1$. If it takes a right+left step, then $\alpha_k = -1$.

So far we have only considered the fast version of our interpolation algorithm. The numerical stability of this algorithm is enhanced via pivoting and iterative refinement [3]. The superfast version that we presented in [5] is based on a divide-and-conquer approach in which at the lowest interpolation level the fast interpolation algorithm is used. Therefore, at this level the accuracy of the computed solution can be improved via pivoting and iterative refinement. After combining solutions of a lower level into a solution of the next level, one can again use iterative refinement. Another technique, explained in detail in [5], postpones some "difficult" interpolation conditions until the very end of the algorithm. All these stabilizing techniques increase the accuracy of the solution to the interpolation algorithm and hence the accuracy of the computed determinant. However, one important difference with the solution of structured linear systems is the fact that we cannot use iterative refinement at the very end of the algorithm to increase the accuracy of the computed determinant. It is an open question if a similar procedure exists for refining approximations for a determinant. As the length of this paper is limited, we refer the reader who is interested in more details and numerical examples to a companion paper about the fast and superfast computation of Toeplitz determinants [2].

Acknowledgements

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] M. Fiedler, *Hankel and Loewner matrices*, Linear Algebr. Appl. **58** (1984), 75–95.

[2] P. Kravanja and M. Van Barel, *Coupled Vandermonde matrices and the superfast computation of Toeplitz determinants*, In preparation.

[3] ______, *A fast Hankel solver based on an inversion formula for Loewner matrices*, Linear Algebr. Appl. **282** (1998), no. 1–3, 275–295.

[4] K. Loewner, *Über monotone Matrixfunktionen*, Math. Z. **38** (1934), 177–216.

[5] M. Van Barel and P. Kravanja, *A stabilized superfast solver for indefinite Hankel systems*, Linear Algebr. Appl. **284** (1998), no. 1–3, 335–355, 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.

[6] Z. Vavřín, *Inverses of Löwner matrices*, Linear Algebr. Appl. **63** (1984), 227–236.