FAST AND ACCURATE CON-EIGENVALUE ALGORITHM FOR OPTIMAL RATIONAL APPROXIMATIONS *

T. S. HAUT AND G. BEYLKIN

Abstract. The need to compute small con-eigenvalues and the associated con-eigenvectors of positive-definite Cauchy matrices naturally arises when constructing rational approximations with a (near) optimally small L^{∞} error. Specifically, given a rational function with n poles in the unit disk, a rational approximation with $m \ll n$ poles in the unit disk may be obtained from the mth con-eigenvector of an $n \times n$ Cauchy matrix, where the associated con-eigenvalue $\lambda_m > 0$ gives the approximation error in the L^{∞} norm. Unfortunately, standard algorithms do not accurately compute small con-eigenvalues (and the associated con-eigenvectors) and, in particular, yield few or no correct digits for con-eigenvalues smaller than the machine roundoff. We develop a fast and accurate algorithm for computing con-eigenvalues and con-eigenvectors of positive-definite Cauchy matrices, yielding even the tiniest con-eigenvalues with high relative accuracy. The algorithm computes the mth con-eigenvalue in $\mathcal{O}\left(m^2 n\right)$ operations and, since the con-eigenvalues of positive-definite Cauchy matrices decay exponentially fast, we obtain (near) optimal rational approximations in $\mathcal{O}\left(n\left(\log\delta^{-1}\right)^2\right)$ operations, where δ is the approximation error in the L^{∞} norm. We provide error bounds demonstrating high relative accuracy of the computed con-eigenvalues and the high accuracy of the unit con-eigenvectors. We also provide examples of using the algorithm to compute (near) optimal rational approximations of functions with singularities and sharp transitions, where approximation errors close to machine roundoff are obtained. Finally, we present numerical tests on random (complex-valued) Cauchy matrices to show that the algorithm computes all the con-eigenvalues and con-eigenvectors with nearly full precision.

1. Introduction

We present an algorithm for computing with high relative accuracy the con-eigenvalue decomposition of positive-definite Cauchy matrices,

$$Cu_m = \lambda_m \overline{u_m}, \quad C_{ij} = \frac{\sqrt{\alpha_i}\sqrt{\overline{\alpha_j}}}{1-\gamma_i\overline{\gamma_j}}, \; i,j = 1,\ldots,n, \tag{1.1}$$

where γ_i and α_i are complex numbers and $|\gamma_i| < 1$.The con-eigenvalue λ_m is only defined up to an arbitrary phase, which we choose so that $\lambda_m > 0$. Although the con-eigenvalue decomposition (see e.g. [30]) is less well-known than the eigenvalue decomposition or the singular value decomposition, it arises naturally in constructing optimal approximations using exponentials or rational functions [1, 2, 3, 14, 40, 6, 7]. For example, for a real-valued rational function $f(z)$,

$$f(z) = \sum_{i=1}^{n} \frac{\alpha_i}{z-\gamma_i} + \sum_{i=1}^{n} \frac{\overline{\alpha_i} z}{1-\overline{\gamma_i} z} + \alpha_0, \tag{1.2}$$

we may construct a rational approximation $g(z)$ with m poles and with an error,

$$\max_{x\in[0,1]} \left| f\left(e^{2\pi i x}\right) - g\left(e^{2\pi i x}\right)\right| \approx \lambda_m,$$

by solving the con-eigenvalue problem (1.1) (see Section 2.1 for more detail). Ordering the con-eigenvalues, $\lambda_1 \geq \ldots \geq \lambda_n > 0$, the number of poles m of the approximant $g(z)$ corresponds to the index of the con-eigenvalue λ_m and leads to a near optimal approximation in the L^{∞}-norm with

This research was partially supported by NSF grant DMS-100995 and DOE/ORNL grant 4000038129.

* Published in SIAM J. Matrix Anal. Appl., v.33, 4, pp. 1101-1125.

the error close to λ_m. The form (1.2) ensures that $f\left(e^{2\pi ix}\right)$ is real-valued and periodic; complex-valued functions may also be handled using this form by splitting the real and imaginary parts and performing additional reductions (see [7]).

Current algorithms compute an approximate con-eigenvalue $\widehat{\lambda_m}$ with an error no better than $\left|\lambda_m-\widehat{\lambda_m}\right|/\left|\lambda_1\right|=\mathcal{O}\left(\epsilon\right)$, and an approximate unit con-eigenvector $\widehat{u_m}$ with an error no better than

$$\left\|u_m-\widehat{u_m}\right\|_2=\mathcal{O}\left(\epsilon\right)/\mathrm{absgap}_m,\quad \mathrm{absgap}_m\equiv\min_{p\neq m}\left|\lambda_m-\lambda_p\right|/\left|\lambda_1\right|,$$

where ϵ denotes the machine roundoff. This implies that a computed con-eigenvalue smaller than $\left|\lambda_1\right|\epsilon$ may have few or no correct digits. Hence, in order to obtain a rational approximation with accuracy $\lambda_m\lesssim10^{-7}$, we may be forced to use at least quadruple precision. Since quadruple precision is typically not supported by the hardware, it slows down the computation by an unpleasant factor (between 30 and 100). Another undesirable feature of current algorithms to solve (1.1) is the $\mathcal{O}\left(n^3\right)$ complexity for finding the $m\ll n$ poles of $g(z)$, where n is the original number of poles of $f(z)$.

Although the construction of optimal rational approximations in the L^∞-norm has a long history (starting with the seminal papers [1, 2, 3]), the difficulties mentioned above limit practical applications of such approximations to situations where the problem size is relatively small and a low accuracy is acceptable. In this regard, we view our results as a stepping stone toward a wider use of optimal L^∞-approximations in numerical analysis (see [27]).

We develop a fast and accurate algorithm for con-eigenvalue/con-eigenvector computations of positive-definite Cauchy matrices that addresses both of the difficulties mentioned above. Our algorithm computes the mth con-eigenvalue/con-eigenvector in $\mathcal{O}\left(m^2n\right)$ operations (see Section 5). Since the con-eigenvalues of positive definite Cauchy matrices decay exponentially fast, for a given desired accuracy $\left\|f\left(e^{2\pi ix}\right)-g\left(e^{2\pi ix}\right)\right\|_\infty\approx\delta$, the number of poles m in the approximant $g(z)$ is $\mathcal{O}\left(\log\delta^{-1}\right)$. Therefore, the complexity of our algorithm is $\mathcal{O}\left(n\left(\log\delta^{-1}\right)^2\right)$, i.e., it is essentially linear in the number of original poles n and, thus, is mostly controlled by the number of poles of the final optimal approximation.

The con-eigenvalue algorithm achieves high relative accuracy, i.e., the computed con-eigenvalue $\widehat{\lambda_m}$ satisfies $\left|\lambda_m-\widehat{\lambda_m}\right|/\left|\lambda_m\right|=\mathcal{O}\left(\epsilon\right)$, and the computed unit con-eigenvector $\widehat{u_m}$ satisfies

$$\left\|u_m-\widehat{u_m}\right\|_2=\mathcal{O}\left(\epsilon\right)/\mathrm{relgap}_m,\quad \mathrm{relgap}_m\equiv\min_{l\neq m}\left|\lambda_m-\lambda_l\right|/\left(\lambda_l+\lambda_m\right),$$

(see Theorems 6 and 7 for the exact statement). In contrast to the usual perturbation theory for general matrices, we show that small perturbations of the poles γ_m and residues α_m (determining the Cauchy matrix $C=C(\alpha,\gamma)$ in (1.1)) lead to correspondingly small perturbations in the con-eigenvalues and con-eigenvectors, as long as the poles are well separated in a relative sense and are not too close to the unit circle.

In many applications, the function $f\left(e^{2\pi ix}\right)$ has sharp transitions, so that the poles are clustered close to the unit circle and each other. In such cases, it is natural to maintain the poles of $f\left(z\right)$ in the form $\gamma_j=\exp\left(-\tau_j\right)$, where $\mathcal{R}e\left(\tau_j\right)>0$ and $0\leq\mathcal{I}m\left(\tau_j\right)<2\pi$, so that $\mathcal{R}e\left(\tau_j\right)$ are well-separated in a relative sense. The reduction algorithm produces new poles of the same form, where even the smallest exponents are computed with high relative accuracy. This allows us to develop a numerical calculus that includes functions with singularities and sharp transitions. We address this issue further in Section 3.

Our approach is inspired by papers [20, 23, 18, 15, 29], which develop algorithms and theory for highly accurate SVDs of certain structured matrices. Generally speaking, high relative accuracy is achieved when it is possible to avoid catastrophic cancellation resulting from subtracting two close floating point numbers (when the outcome of such cancellation is significant relative to the final result). We refer to [16] for a comprehensive analysis of when efficient and accurate algorithms are possible using floating point arithmetic. Classes of matrices for which highly accurate SVD or eigenvalue algorithms exist include bi-diagonal matrices [19, 13, 26], acyclic matrices [21], graded positive-definite matrices [20], scaled diagonally dominant matrices [4], totally positive matrices [31], certain indefinite matrices [36], and Cauchy matrices (as well as, more generally, matrices with displacement rank one) [15]. For such matrices, recent algorithmic advances (see [24, 25]) make the cost of achieving high relative accuracy comparable to that of alternative (and less accurate) SVD methods.

The con-eigenvalue algorithm considered here is based on computing the eigenvalue decomposition of the product, $\overline{C}C$, of positive-definite Cauchy matrices $\overline{C}$ and C, and is similar to the algorithm in [17] for the generalized eigenvalue decomposition, as well as the algorithm in [23] for the product SVD decomposition. We also rely on the algorithm in [15] for computing, with high relative accuracy, the Cholesky decomposition (with complete pivoting) $C = (PL)\,D^2\,(PL)^*$ of a positive-definite Cauchy matrix C. However, since we are interested in computing only con-eigenvalues of some approximate size δ, we stop Demmel's Cholesky algorithm once the diagonal elements D_{ii} are small with respect to δ and the desired precision. Since the diagonal elements D_{ii} decay exponentially fast, this allows us to accurately compute con-eigenvalues of size δ (and the associated con-eigenvectors) in $\mathcal{O}\left(n\left(\log\delta^{-1}\right)^2\right)$ operations. We also modify the Cholesky decomposition algorithm in [15] to yield high relative accuracy for Cauchy matrices $C_{ij} = \sqrt{\alpha_i}\sqrt{\overline{\alpha_j}}/\left(1-\gamma_i\overline{\gamma_j}\right)$, with $\gamma_i = \exp\left(-\tau_j\right)$, where the real parts of the exponents, $\mathcal{R}e(\tau_j)$, may be extremely small in magnitude. We observe that the error bounds developed in [23] are not applicable to our problem since the condition number of a Cauchy matrix cannot be appreciably reduced by scaling the rows and columns. In contrast, the error bounds presented in this paper yield high relative accuracy for all the computed con-eigenvalues larger than δ (and high accuracy for the con-eigenvectors), as long as L is well-conditioned, and the relative gap between the con-eigenvalues is not too small (we have always observed this to hold in practice). In particular, if δ is chosen small enough, the full con-eigenvalue decomposition is obtained with high relative accuracy. The derivation of our error bounds makes crucial use of the component-wise perturbation theory developed in [20] for the singular vectors of graded matrices (see also [34]), as well as the component-wise error analysis in [20] and [33] for the one-sided Jacobi method. We also use the error analysis given in [29] for the Householder QR method. We note that although our error estimates are much more pessimistic than what we observe in practice, they provide a framework for understanding the high accuracy of the con-eigenvalue algorithm of this paper. In order to limit the size of this paper, proofs can be found in its online version [28].

It has been an established practice, in both numerical analysis and signal processing, to use L^2-type methods for representing functions. On the other hand, it has been understood for some time that nonlinear approximations may be far superior in achieving high accuracy with a minimal number of terms (see e.g., [35]). However, in spite of many interesting results (see e.g., [32, 37, 14, 38, 39, 40, 6, 8, 22]), the widespread use of nonlinear approximations has been limited by a lack of efficient and accurate algorithms for computing them (particularly for functions with sharp changes or singularities). Our algorithms provide the necessary tools for computing optimal nonlinear approximations via rational functions, and come with guaranteed accuracy bounds. We believe that these new accurate algorithms may greatly extend the practical use of L^∞ approximations in numerical analysis (see [27]) and signal processing (see [5]).

In Section 2.1 we describe the reduction problem for rational functions, and connect its solution to a con-eigenvalue problem for positive definite Cauchy matrices. We then present new algorithms for solving the con-eigenvalue problem with high relative accuracy. We follow up in Section 3 with examples of using the reduction algorithm to construct and use optimal rational approximations for functions with singularities and sharp transitions. In Section 4 we verify the accuracy of the con-eigenvalue algorithm by comparing the con-eigenvalue decomposition of randomly generated Cauchy matrices with that obtained via standard algorithms in extended precision. In Section 5, we provide error bounds that demonstrate the con-eigenvalue algorithm achieves high relative accuracy and that the con-eigenvalue decomposition is stable with respect to small perturbations of the parameters defining the Cauchy matrix. Finally, Section 6 compares the reduction algorithm of this paper with other algorithms in the literature for constructing optimal rational approximations. For the convenience of the reader we also provide relevant background material in Section 7.

2. Accurate con-eigenvalue decomposition (an informal derivation)

2.1. **Constructing optimal rational approximations via a con-eigenvalue problem.** In order to motivate our con-eigenvalue algorithm, let us explain how the accurate computation of small con-eigenvalues and associated con-eigenvectors allows us to construct optimal rational approximations.

We consider an algorithm to find a rational approximation $r(e^{2\pi ix})$ to $f(e^{2\pi ix})$ in (1.2) with a specified number of poles and with a (nearly) optimally small error in the L^{∞}-norm. The algorithm is based on a theorem of Adamyan, Arov, and Krein (referred to below as the AAK Theorem) [3]. We note that the formulation given below in terms of a con-eigenvalue problem is similar to the approach taken in [14] and [6].

Given a target accuracy δ for the error in the L^{∞}-norm, the steps for computing the rational approximant $r(z)$,

$$r(z)=\sum_{i=1}^{m}\frac{\beta_i}{z-\eta_i}+\sum_{i=1}^{m}\frac{\overline{\beta_i}z}{1-\overline{\eta_i}z}+\alpha_0,$$

are as follows:

(1) Compute a con-eigenvalue $0<\lambda_m\leq\delta$ and corresponding con-eigenvector u of the Cauchy matrix $C_{ij}=C_{ij}(\gamma_i,\alpha_j)$,

$$Cu=\lambda_m\overline{u},\ \text{ where }\ u=\left(\begin{array}{c}u_1\\u_2\\\vdots\\u_n\end{array}\right),\quad C_{ij}=\frac{a_ib_j}{x_i+y_j},\ \ i,j=1,\ldots,n, \tag{2.1}$$

and $a_i=\sqrt{\alpha_i}/\gamma_i$, $b_j=\sqrt{\overline{\alpha_j}}$, $x_i=\gamma_i^{-1}$, $y_j=-\overline{\gamma_j}$. The con-eigenvalues of C are labeled in non-increasing order, $\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_n$.

(2) Find the (exactly) m zeros η_j in the unit disk of the function

$$v(z)=\frac{1}{\lambda_m}\sum_{i=1}^{n}\frac{\sqrt{\overline{\alpha_i}}\,\overline{u_i}}{1-\overline{\gamma_i}z}. \tag{2.2}$$

The fact that there are exactly m zeros in the unit disk, corresponding to the index m of the con-eigenvalue λ_m, is a consequence of the AAK theorem. The poles of $r(z)$ are given by the zeros η_j of $v(z)$.

(3) Find the residues β_m of $r(z)$ by solving the $m\times m$ linear system

$$\sum_{i=1}^{m}\frac{1}{1-\eta_i\overline{\eta_j}}\beta_i=\sum_{i=1}^{n}\frac{\alpha_i}{1-\gamma_i\overline{\eta_j}}. \tag{2.3}$$

The L^{∞}-error of the resulting rational approximation $r(e^{2\pi ix})$ satisfies $\left\|f-r\right\|_{\infty}\approx\lambda_m$, and is close to the best error in the L^{∞}-norm achievable by rational functions with no more than m poles in the unit disk. Hence, we are led to the problem of computing, to high relative accuracy, small con-eigenvalues and the associated con-eigenvectors of positive-definite Cauchy matrices.

In many applications it is natural (and advisable) to maintain the poles γ_j in the form $\gamma_j=\exp\left(-\tau_j\right)$ (see e.g., [6, 8]). As we explain in Section 3, this is particularly important if the function $f(e^{2\pi ix})$ has singularities or sharp transitions. The advantage of this form is that, on a logarithmic scale, the nodes are well separated (i.e., $\mathcal{R}e\left(\tau_j\right)$ are well-separated in a relative sense). In such cases, our algorithm computes the new poles $\eta_i=\exp\left(-\zeta_i\right)$ with nearly full precision in the exponents ζ_i, i.e., $\left|\hat{\zeta}_i-\zeta_i\right|/\left|\zeta_i\right|$ is close to machine precision even if ζ_i is close to zero.

Remark 1. In practice, finding the new poles η_i using the formula for $v(z)$ in (2.2) is ill-advised, since evaluating $v(z)$ in this form could result in loss of significant digits through catastrophic cancellation. Indeed, it turns out (see [6, Section 6] and [27]) that the values of the con-eigenvector components satisfy $u_i = \sqrt{\alpha_i} v(\gamma_i)$, $i = 1, \dots, n$. It then follows that the sum (2.2) must suffer cancellation of about $\log_{10}\left(\lambda_m^{-1}\right)$ digits if $v(\gamma_i)$ and $v(z)$ are of comparable size (note that λ_m controls the approximation error and, thus, is necessarily small). On the other hand, the function values $v(\gamma_i) = u_i/\sqrt{\alpha_i}$, $i = 1, \dots, n$, along with the n poles $1/\overline{\gamma_i}$ of $v(z)$, completely determine (2.2). Since the poles γ_i of $f(z)$ are often close to the poles η_i of $r(z)$, we have observed that evaluating $v(z)$ by using rational interpolation via continued fractions with the known values $v(\gamma_i)$ allows us to obtain the new poles η_i with nearly full precision. In particular, an approximation $\widetilde{v}(z)$ to $v(z)$ is computed via continued fractions,

$$\widetilde{v}(z) = \frac{a_1}{1 + a_2 (z - \gamma_1) / (1 + a_3 (z - \gamma_2) / (1 + \cdots))}, \tag{2.4}$$

where the coefficients a_j are determined from the interpolation conditions $\widetilde{v}(\gamma_i) = v(\gamma_i)$. If the poles γ_i are given in the form $\gamma_i = \exp(-\tau_i)$, we find that Newton's method on $\widetilde{v}(\exp(-\eta))$ yields the new poles $\eta_i = \exp(-\zeta_i)$ with nearly full relative accuracy even when $\mathrm{Re}(\zeta_i) \ll 1$; see Section 3 for more details (achieving high relative accuracy also requires slightly modifying the recursion formulas for the continued fraction coefficients a_i). A more detailed description of the root-finding algorithm may be found in [27].

2.2. Accurate con-eigenvalue decompositions of positive-definite matrices with RRDs.

The con-eigenvalue problem for a positive-definite Cauchy matrix $C_{ij} = a_i b_j / (x_i + y_j)$ reduces to an eigenvalue problem,

$$\overline{C} C u = \lambda \overline{C} \bar{u} = |\lambda|^2 u. \tag{2.5}$$

We first discuss a somewhat more general problem of computing accurate eigenvalues and eigenvectors of matrices of the form $\overline{A} A$, where we assume that A has a factorization $A = X D^2 X^*$, with X a (well-conditioned) $n \times m$ matrix ($m \leq n$) and D an $m \times m$ diagonal matrix with positive, non-increasing diagonal entries. The rectangular form of the factorization, $m \leq n$, will be important in the sequel.

Let us define the $m \times m$ matrix $G = D\left(X^{\mathrm{T}} X\right) D$, and consider its SVD, $G = W \Sigma V^*$. Then $G^* G = V \Sigma^2 V^*$, and the ith right singular vector ($1 \leq i \leq m$), $v_i = V(:, i)$, satisfies $\left(D X^* \overline{X} D\right)\left(D X^{\mathrm{T}} X D\right) v_i = \Sigma_{ii}^2 v_i$. It then follows that $z_i = X D v_i$ is an eigenvector of $A \overline{A}$ with eigenvalue Σ_{ii}^2, since

$$\begin{aligned} A \overline{A} z_i &= \left(X D^2 X^*\right)\left(\overline{X} D^2 X^{\mathrm{T}}\right) z_i = \\ &= X D \left(D X^* \overline{X} D\right)\left(D X^{\mathrm{T}} X D\right) v_i = \Sigma_{ii}^2 X D v_i = \Sigma_{ii}^2 z_i. \end{aligned}$$

and, thus, $\overline{z_i} = \overline{X} D \overline{v_i}$ is an eigenvector of $\overline{A} A$. To summarize: given the decomposition $A = X D^2 X^*$, an eigenvector z_i ($i \leq m$) of $\overline{A} A$ is given by $\overline{z_i} = \overline{X}\left(D \overline{v_i} \Sigma_{ii}^{-1/2}\right)$, where v_i is the ith right singular vector of the $m \times m$ matrix $G = D\left(X^{\mathrm{T}} X\right) D$. Here Σ_{ii} is the ith singular value of G, and the ith con-eigenvalue of A. Let us now present an algorithm for accurately computing the con-eigenvalues and con-eigenvectors of A (its derivation also relies on the background material collected in Section 7).

Algorithm 1 ConEig_RRD (X, D) computes accurate con-eigenvalue decomposition of XD^2X^*. Input: rank-revealing factors X and D (of dimensions $n \times m$ and $m \times m$), where the diagonal of $D > 0$ is decreasing. Output: m con-eigenvalues/con-eigenvectors of XDX^*, contained in Σ and T.

```
$(\Sigma, T) \leftarrow$ ConEig_RRD $(X, D)$
    1. Form $G = D\,(X^{\mathrm{T}}X)\,D$
    2. Compute QR factors $(Q, R) \leftarrow$ Householder_QR of $G$ ($G = QR$), with optional pivoting (see Section 7.3)
    3. Compute the SVD factors $(U_l, \Sigma, U_r) \leftarrow$ Jacobi$(R)$ of $R$ ($R = U_l \Sigma U_r^*$), using one-sided Jacobi, applied from the left (see Section 7.4)
    4. Compute $R_1 = D^{-1}RD^{-1}$, $X_1 = D^{-1}U_l\Sigma^{1/2}$, and $Y_1 = R_1^{-1}X_1$ (see (2.6) below)
    5. Form the matrix of con-eigenvectors $T = \overline{XY_1}$, and output con-eigenvalues $\Sigma$ and con-eigenvectors $T$
```

Importantly, for Cauchy matrices ($A = C$) the elements of D decay exponentially fast, and it would appear that computing the con-eigenvectors $\overline{z_i} = \overline{X}D\overline{v_i}/\Sigma_{ii}^{1/2}$ might lead to wildly inaccurate results even if the right singular vector of G, v_i, is computed accurately. However, as we show in Section 5, Algorithm 1 achieves high accuracy despite the extreme ill-conditioning of D. The key reason is that the right singular vector v_i, corresponding to the singular value Σ_{ii}, scales like $|v_i(j)| \leq c_V \min\left(D_{jj}/\Sigma_{ii}^{1/2}, \Sigma_{ii}^{1/2}/D_{jj}\right)$, and the computed singular vector $\widehat{v_i}$ is accurate relative to the scaling in D and Σ in the sense that

$$|v_i(j) - \widehat{v_i}(j)| \leq \min\left\{\frac{D_{jj}}{\sqrt{\Sigma_{ii}}}, \frac{\sqrt{\Sigma_{ii}}}{D_{jj}}\right\}\mathcal{O}(\epsilon).$$

For Cauchy matrices, the quantity $\min\left(D_{jj}/\Sigma_{ii}^{1/2}, \Sigma_{ii}^{1/2}/D_{jj}\right)$ decreases exponentially fast away from the diagonal $i = j$.

Let us give an informal explanation of the reasons why Algorithm 1 yields accurate results. As discussed in Section 7.3, the QR Householder algorithm computes an accurate rank-revealing decomposition of $G = QR$. It turns out (see the online version [28, Lemma 11]) that R may be factored as $R = D^2R_0$, where R_0 is graded relative to D in the sense that$\left\|DR_0D^{-1}\right\|$ and $\left\|DR_0^{-1}D^{-1}\right\|$ are not too large, as long as the n leading principal minors of $X^{\mathrm{T}}X$ are well-conditioned. Therefore, from the discussion in Section 7.4 (see in particular Theorem 10), the one-sided Jacobi algorithm computes the ith left singular vector u_i of R accurately relative to the scaling $\min\left\{D_{jj}/\Sigma_{ii}^{1/2}, \Sigma_{ii}^{1/2}/D_{jj}\right\}$. It follows that $D^{-1}u_i\Sigma_{ii}^{1/2}$ may also be computed accurately. Finally, since the ith right singular vector v_i of R (and G) satisfies

$$\begin{aligned} Dv_i\Sigma_{ii}^{-1/2} &= DR^{-1}u_i\Sigma_{ii}^{1/2} \\ &= \left(DR_0D^{-1}\right)^{-1}\left(D^{-1}u_i\Sigma_{ii}^{1/2}\right), \end{aligned} \tag{2.6}$$

the con-eigenvector $\overline{z_i} = \overline{X}\left(D\overline{v_i}\Sigma_{ii}^{-1/2}\right)$ may be computed accurately, as long as DR_0D^{-1} is computed accurately and is well-conditioned (we show this is the case if n leading principal minors of $X^{\mathrm{T}}X$ are well-conditioned). The last step in Algorithm 1 uses the approach in [25] for computing highly accurate right singular vectors via solving a triangular linear system of equations.

Remark 2. To obtain optimal rational approximations (see Section 2.1), we need to compute small con-eigenvalues (and the associated con-eigenvectors) of Cauchy matrices of the slightly different form, $C_{ij} = \sqrt{\alpha_i}\sqrt{\overline{\alpha}_j}/\left(1 - \gamma_i\overline{\gamma_j}\right)$, i.e., with $a_i = \sqrt{\alpha_i}/\gamma_i$, $b_j = \sqrt{\overline{\alpha}_j}$, $x_i = \gamma_i^{-1}$, and $y_j = -\overline{\gamma_j}$. The same reasoning as in [15] shows that the Cholesky computation of C (see Section 7.2) is performed with high relative accuracy, as long as the differences $\gamma_j^{-1} - \overline{\gamma_i}$ are computed with high relative accuracy. As explained in the next section, $\gamma_j^{-1}-\overline{\gamma_i}$ may be accurately computed if γ_i is of the form $\gamma_i=\exp\left(-\tau_i\right)$, where the exponents τ_i are known accurately (see Section 3 for examples).

Remark 3. Computing the normalized eigenvector u via (2.5) determines the con-eigenvector, the solution of (2.1), only up to an unknown phase factor $e^{-i\phi/2}$. Indeed, given any solution λ and u of (2.5) and an arbitrary phase factor $e^{-i\phi}$, it is easy to see that $\lambda e^{-i\phi}$ and $ue^{-i\phi/2}$ also satisfy (2.1). Let us now determine the phase ϕ so that the con-eigenvalues λ are positive. To do so, we compute the usual inner product $\left(C\left(ue^{-i\phi/2}\right),ue^{-i\phi/2}\right)=\lambda\left(\overline{u}e^{i\phi/2},ue^{-i\phi/2}\right)$ and choose ϕ so that $\lambda>0$. Since C is a positive-definite matrix, it follows that $\left(\overline{u}e^{i\phi/2},ue^{-i\phi/2}\right)>0$. From this we obtain the phase factor as $e^{i\phi}=\left(u,\overline{u}\right)/\left|\left(u,\overline{u}\right)\right|$.

2.3. **Accurate con-eigenvalue decompositions of positive-definite Cauchy matrices.** If $A=C$ is a positive-definite Cauchy matrix, then the modified GECP algorithm in [15] computes the Cholesky decomposition $C=\left(PL\right)D^2\left(PL\right)^*$ with high relative accuracy (see Section 7.1). Therefore, Algorithm 1 for the eigenvalue problem of $\overline{C}C$ may be used, with $X=PL$, to compute all the eigenvalues and eigenvectors (and, therefore, the con-eigenvectors and con-eigenvalues of C).

For our purposes, we are only interested in computing a single con-eigenvector with associated con-eigenvalue of approximate size δ (see Section 2.1). However, the diagonal elements of D may be many orders of magnitude smaller than δ, and it is then natural to expect that, by computing a partial Cholesky decomposition of C, we may obtain the ith con-eigenvector in much fewer than $\mathcal{O}\left(n^3\right)$ operations. In this case, we stop Demmel's algorithm for the Cholesky decomposition of C once the diagonal elements D_{ii}^2 are small with respect to the product of δ^2 and the machine round-off ϵ, that is, as soon as $D_{mm}^2\leq\delta^2\epsilon$ for some m (notice that complete pivoting ensures that the diagonal elements D_{ii} are non-increasing). We then obtain $C\approx\widetilde{C}=\left(\widetilde{P}\widetilde{L}\right)\widetilde{D}^2\left(\widetilde{P}\widetilde{L}\right)^*$, where $\widetilde{P}$ is an $m\times n$ matrix, $\widetilde{L}$ is an $n\times m$ matrix and $\widetilde{D}$ is a diagonal $m\times m$ matrix. Algorithms 2 and 3 contain pseudo-code for computing $\widetilde{L}$, $\widetilde{D}$, and $\widetilde{P}$. In the pseudo-code $I\left(n,m\right)$ denotes the first $m\leq n$ columns of the $n\times n$ identity matrix.

Algorithm 2 Pivot_Order$\,(a,b,x,y,\delta)$ pre-computes pivot order for Cholesky factorization of $n\times n$ positive-definite Cauchy matrix $C_{ij} = a_i b_j/\left(x_i+y_j\right)$. Input: a, b, x, and y defining $C_{ij} = a_i b_j/\left(x_i+y_j\right)$, and target size δ of con-eigenvalue. Output: correctly pivoted vectors a, b, x, and y, truncation size m, and $m\times n$ permutation matrix $\widetilde{P}$

```
                      $\left(a,b,x,y,\widetilde{P},m\right) \leftarrow$ Pivot_Order$\,(a,b,x,y,\delta)$
Form vector $g_i := a_i b_i/(x_i+y_i)$, $i=1,\ldots,n$
Set cutoff for GECP termination: $\eta := \epsilon\delta^2$
Initialize permutation matrix ($n\times n$ identity): $\widetilde{P} = I\,(n,n)$
Compute correctly pivoted vectors:
 $m := 1$
    while $|g\,(m)| \geq \eta$ or $m = n-1$
       Find $m \leq l \leq n$ such that $|g(l)| = \max|g\,(m:n)|$
       Swap elements:
        $g(l) \leftrightarrow g(m)$, $x(l) \leftrightarrow x(m)$ , $y(l) \leftrightarrow y(m)$
        $a(l) \leftrightarrow a(m)$,$b(l) \leftrightarrow b(m)$
       Swap rows of permutation matrix:
        $\widetilde{P}(l,:) \leftrightarrow \widetilde{P}(m,:)$
       Update diagonal of Schur complement:
        $g(m+1:n) :=$
    $\left(x\,(m+1:n)-x(m)\right)/\left(y\,(m+1:n)-y(m)\right)g(m+1:n)$
       Increment iteration count:
        $m := m+1$
Output $a,b,x,y,\widetilde{P}\,(1:m,n)\,,m$
```

Algorithm 3 Cholesky_Cauchy$\,(x,y,a,b,\delta)$ computes partial Cholesky factorization of positive-definite Cauchy matrix $C_{ij} = a_i b_j/\left(x_i+y_j\right)$. Input: a, b, x, and y defining $C_{ij} = a_i b_j/\left(x_i+y_j\right)$, and target size δ of con-eigenvalue. Output: $n\times m$ matrix $\widetilde{L}$, $m\times m$ matrix $\widetilde{D}$, and permutation $m\times n$ matrix $\widetilde{P}$ in partial Cholesky factorization.

```
                      $\left(\widetilde{L},\widetilde{D},\widetilde{P}\right) \leftarrow$ Cholesky_Cauchy$\,(a,b,x,y,\delta)$
Compute pivoted vectors and matrix size $m$ (Algorithm 2):
  $\left(a,b,x,y,\widetilde{P},m\right) \leftarrow$ Pivot_Order$\,(a,b,x,y,\delta)$
Initialize generators:
  $\alpha := a$, $\beta := b$
Compute first column of Schur complement:
  $G\,(:,1) := \alpha * \beta/\left(x+y\right)$
  for $k = 2,m$
       Update generators:
          $\alpha\,(k:n) := \alpha\,(k:n) * \left(x\,(k:n) - x\,(k-1)\right)/\left(x\,(k:n)+y\,(k-1)\right)$
          $\beta\,(k:n) := \beta\,(k:n) * \left(y\,(k:n) - y\,(k-1)\right)/\left(y\,(k:n)+x\,(k-1)\right)$
       Extract $k$th column for Cholesky factors:
          $G\,(k:n,k) := \alpha\,(k:n) * \beta\,(k:n)\,/\left(x\,(k:n)+y\,(k:n)\right)$
Output partial Cholesky factors:
  $\widetilde{D} = $ diag$\left(G(1:n,1:m)\right)^{1/2}$, $\widetilde{L} = $ tril$\left(G(1:n,1:m)\right)\widetilde{D}^{-2} + I\,(n,m)$, $\widetilde{P}$
```

Once the partial Cholesky decomposition $C \approx \widetilde{C} = \left(\widetilde{P}\widetilde{L}\right) \widetilde{D}^2 \left(\widetilde{P}\widetilde{L}\right)^*$ is computed, Algorithm 1 for the eigenvalue problem of $\overline{\widetilde{C}}\widetilde{C}$ may then be used, with $X = \widetilde{P}\widetilde{L}$ and $D = \widetilde{D}$, to compute accurate con-eigenvalues and con-eigenvectors of $\widetilde{C}$ (see Theorem 7). Since the con-eigenvalues decay exponentially fast, the complexity of this algorithm is $\mathcal{O}\left(n\left(\log(\delta\epsilon)^{-1}\right)^2\right)$ operations. Therefore, when used in the reduction procedure outlined in Section 2.1, the near optimal rational approximation may be obtained by computing the SVD of a matrix that is roughly twice the size of the optimal number of poles. The pseudo-code is given in Algorithm 4.

Algorithm 4 Con_Eigvector(a,b,x,y,δ) computes accurate con-eigenvalue decomposition of positive-definite Cauchy matrix $C_{ij} = a_i b_j/(x_i+y_j)$. Input: a, b, x, and y defining $C_{ij} = a_i b_j/(x_i+y_j)$, and target size δ of con-eigenvalue. Output: con-eigenvalues lager than δ, and associated con-eigenvectors.

```
$(\Sigma, T) \leftarrow$ Con_Eigvector$(a,b,x,y,\delta)$
    1. Compute partial Cholesky factors $(L,D,P) \leftarrow$
    Cholesky_Cauchy$(a,b,x,y,\delta)$ (Algorithm 3) and set $X = PL$
    2. Compute con-eigenvalues and con-eigenvectors $(\Sigma,T) \leftarrow$
    ConEig_RRD$(X,D)$ using Algorithm 1
    3. Select largest $l$ such that $\Sigma_{ll} \geq$
    $\delta$ and output $\Sigma(1:l,1:l)$, $T(1:n,1:l)$
```

Remark 4. In applications involving functions $f\left(e^{2\pi i x}\right)$ with singularities or sharp transitions, the poles γ_i are given in the form $\gamma_i = \exp(-\tau_i)$, where $\mathcal{R}e\tau_j > 0$ and $0 \leq \mathcal{I}m\tau_j < 2\pi$ and the exponents τ_i are known with high relative accuracy. Indeed, this form naturally arises either via a discretization of an integral (see [6, 8]) or as a result of an intermediate computation as in [27]. This leads us to modify Algorithms 2 and 3 so that high relative accuracy is achieved for poles of this form. In particular, we modify formulas (7.7), (7.8) and (7.9) in Section 7. For example, the formula for $\alpha_i^{(k)}$ in (7.9) involves computing

$$\frac{x_j - x_{k-1}}{x_j + y_{k-1}} = \frac{\gamma_j^{-1} - \gamma_{k-1}^{-1}}{\gamma_j^{-1} - \overline{\gamma_{k-1}}} = \frac{1-\exp\left(-\tau_j + \tau_{k-1}\right)}{1-\exp\left(-\tau_j - \overline{\tau_{k-1}}\right)}.$$

The simple modification is to use the Taylor expansion $1 - \exp(z) \approx z + z^2/2 + \dots$ if $|z|$ is small. The other formulas in (7.7), (7.8) and (7.9) are modified in a similar fashion, allowing the LDU factorization of C to be computed with high relative accuracy.

In Section 3, we consider a case where the absolute values of many poles agree with 1 to twelve digits (i.e., the poles γ_i satisfy$|\gamma_i| \approx 0.999999999999xxxx$).

3. Examples of optimal rational approximations

In this section, we consider some applications of the reduction algorithm.

3.1. **Optimal rational approximations of functions with singularities.** Using the reduction algorithm, as well as tools developed in [6, 8], we construct a (near) optimal rational approximation of a (piecewise smooth) function f with a finite number of isolated integrable singularities. For simplicity, we assume that singularities of f are at two points, 0 and x_0.

Performing integration by parts L times on the expression for the Fourier coefficients,

$$\hat{f}_n = \int_0^1 f(x)e^{2\pi i n x}dx = \int_0^{x_0} f(x)e^{2\pi i n x}dx + \int_{x_0}^1 f(x)e^{2\pi i n x}dx,$$

we obtain

$$\widehat{f}_n = h_n + \frac{(-1)^L}{(2\pi in)^L}\int_0^{x_0} f^{(L)}(x)e^{2\pi inx}dx + \frac{(-1)^L}{(2\pi in)^L}\int_{x_0}^1 f^{(L)}(x)e^{2\pi inx}dx,$$

where

$$h_n = \sum_{p=1}^{L} \frac{(-1)^p}{(2\pi in)^p}\left(e^{2\pi inx_0}F^{(p-1)}(x_0) + F^{(p-1)}(0)\right),$$

$F^{(p)}(x) = f^{(p)}(x^+) - f^{(p)}(x^-)$ and x^+, x^- indicate directional limits. As the first step in constructing a (near) optimal rational approximation to f, we subtract the leading L terms of the asymptotic expansion of $\widehat{f}_n$ and consider $g_n = \widehat{f}_n - h_n$. Since g_n decays like $O\left(1/n^{L+1}\right)$, it is sufficient to use the algorithm in [6, 8] to construct an approximation

$$\left|g_n - \sum_{m=1}^{M} w_m e^{-\mu_m n}\right| \leq \epsilon, \quad n \geq 1. \tag{3.1}$$

This algorithm requires quadruple precision for computing small singular values of a Hankel matrix but, due to the fast decay of g_n, the matrix is small so that the computational cost is insignificant. An alternative method for obtaining (3.1) based on rational representations of B-splines requires only double precision and will appear elsewhere [11]. For h_n we use a discretization of the integral representation for $1/n^p$ in [8] to obtain

$$\left|\frac{1}{n^p} - \sum_{m=-M_1}^{M_2} a_{m,p}e^{-\tau_m n}\right| \leq \epsilon, \quad 1 \leq p \leq L, \quad 1 \leq n, \tag{3.2}$$

where $\tau_m = e^{hm}$, $a_{m,p} = \frac{h}{(p-1)!}e^{phm}$ and h is the step size used in the discretization. Results in [8] imply that there are at most $\mathcal{O}\left(\left(\log\epsilon^{-1}\right)^2\right)$ terms in the approximation of $1/n^p$ for a given accuracy ϵ, for all $n \geq 1$. Note that when $m < 0$ the nodes $\gamma_m = e^{-e^{hm}} \approx 1 - e^{hm}$ are very close to one.

Thus, we arrive at

$$\left|h_n - \sum_{m=-M_1}^{M_2} a_m e^{-(\tau_m + 2\pi ix_0)n} - \sum_{m=-M_1}^{M_2} b_m e^{-\tau_m n}\right| \leq 2\epsilon, \tag{3.3}$$

where

$$a_m = \sum_{p=1}^{L} \frac{1}{(-2\pi i)^p}F^{(p-1)}(x_0)\,a_{m,p}, \quad b_m = \sum_{p=1}^{L} \frac{1}{(-2\pi i)^p}F^{(p-1)}(0)\,a_{m,p}.$$

Combining the approximations (3.1) and (3.3), we obtain the suboptimal approximation

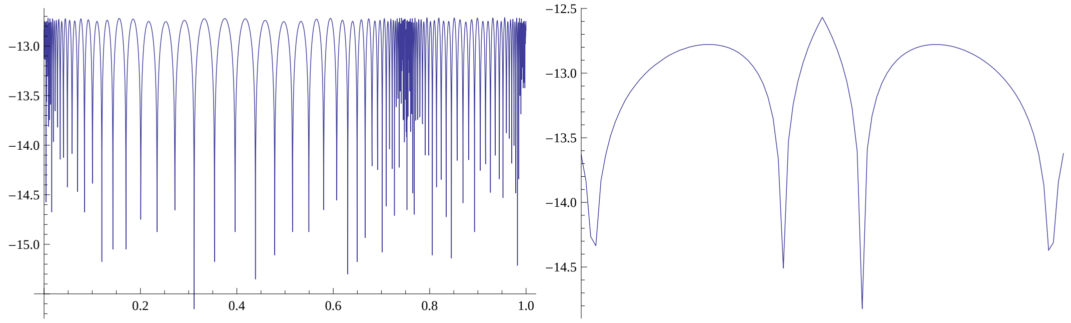
$$\left|\widehat{f}_n - \sum_{m=1}^{M} w_m e^{-\mu_m n} - \sum_{m=-M_1}^{M_2} a_m e^{-(\tau_m + 2\pi ix_0)n} - \sum_{m=-M_1}^{M_2} b_m e^{-\tau_m n}\right| \leq 3\epsilon, \tag{3.4}$$

where the number of terms is excessive (for the accuracy 3ϵ). We now use the reduction algorithm on (3.4) to obtain a nearly optimal number of terms to approximate the Fourier coefficients f_n for $n \geq 1$. This, in turn leads to a near optimal rational approximation to $f(x)$ with a nearly equioscillating error.

As an example, we apply this procedure to

$$f(x) = \begin{cases} \sin(4/3\pi x), & 0 \leq x \leq 3/4 \\ 0 & 3/4 < x \leq 1 \end{cases} \tag{3.5}$$

Choosing the parameters $M_1 = 200$, $M_2 = 10$, and $h = .316707$ in (3.4) (see [8] for how to select the parameters) yields a sub-optimal approximation containing 426 pairs of conjugate-reciprocal poles $\gamma_j = e^{-\tau_j}$, which approximates $f(x)$ in the L^∞ norm with error $\approx 5 \times 10^{-14}$. We note that many



(a) (b)

FIGURE 3.1. (a) Error of the rational approximation to $f(x)$ in (3.5). (b) A zoom on a neighbourhood around one of the singularities $x \in \left(3/4 - 10^{-12}, 3/4 + 10^{-12}\right)$.

of the poles are extremely close to the unit disk (the magnitudes $|\gamma_i| \approx .999999999999xxxx$ of over a dozen poles agree with 1 to twelve digits).

We apply the reduction algorithm using the approximation error $\delta = 10^{-13}$ (thus, the Cholesky decomposition algorithm 3 is truncated once the diagonal elements are smaller than $\epsilon\delta^2$, where ϵ denotes the machine roundoff). As explained in Remark 4, Algorithms 2 and 3 are modified to accurately compute the partial Cholesky decomposition for poles in the form $\gamma_j = e^{-\tau_j}$. After applying the reduction algorithm with approximation error $\delta = 10^{-13}$, the resulting rational approximation contains 92 pairs of conjugate-reciprocal poles (i.e., about 46 poles per singularity). The resulting error is shown in Figure 3.1.

We note that the only step of the reduction procedure where quadruple precision is used is in computing the residues β_j (see Step 3 of Section 2.1). However, using the techniques described in the background Section 7.2 to factor the $m \times m$ Cauchy matrix, this step takes only $\mathcal{O}\left(m^2\right)$ operations, and so does not impact the overall speed of the algorithm (recall that m denotes the number of reduced poles).

We find that the exponents, η_i, of the near optimal poles $\zeta_i = \exp\left(-\eta_i\right)$ are computed with high relative accuracy, i.e.,

$$|\mathrm{Re}\left(\eta_i\right) - \mathrm{Re}\left(\widehat{\eta_i}\right)| \leq |\mathrm{Re}\left(\eta_i\right)|\,\delta_1, \quad |\eta_i - \widehat{\eta_i}| \leq |\eta_i|\,\delta_2,$$

where $\delta_1 \leq 1.48 \times 10^{-13}$ and $\delta_2 \leq 14.87 \times 10^{-13}$. As a gauge we used the poles ζ_i obtained in Mathematica^{TM} via extended precision arithmetic. We note that the real parts of some of the exponents η_i are of size $|\mathrm{Re}\left(\eta_i\right)| \approx 10^{-12}$.

3.2. **Solving viscous Burgers' equation.** In [27] we use the reduction algorithm to solve viscous Burgers' equation,

$$u_t - uu_x = \nu u_{xx}, \;\; u(x,0) = u_0(x), \;\; u(0,t) = u(1,t), \;\; x \in [0,1], \;\; t \geq 0. \tag{3.6}$$

The solution of this equation develops a shock (or a sharp transition) on an interval of size $\mathcal{O}\left(\nu\right)$. We approximate solutions to (3.6) using rational functions of the form

$$u\left(x,t\right) = \sum_{j=1}^{M_0} \frac{\alpha_j\left(t\right)}{e^{-2\pi i x} - \gamma_j\left(t\right)} + \sum_{j=1}^{M_0} \frac{\overline{\alpha_j\left(t\right)}}{e^{2\pi i x} - \overline{\gamma_j\left(t\right)}} + \alpha_0.$$

The key idea is to develop a numerical calculus using the reduction algorithm. Although operators such as multiplication and convolution increase the number of poles in the representation, the reduction algorithm is employed at each stage to keep the number of poles near optimally small. Overall, about 10^6 applications of the reduction algorithm were employed to compute the solutions illustrated below, thus confirming its robustness and efficiency.

Figure 3.2 shows the computed solutions $u(x,h_t j)$ to (3.6), with the viscosity $\nu = 10^{-5}$, the step size h_t and the initial condition $u_0(x) = \sin(2\pi x) + 1/2\sin(4\pi x)$. In our reduction procedure, we used the step size of $h_t = 10^{-5}$ and the error tolerance $\delta = 10^{-9}$ (to match the error of our time discretization). The solution $u(x,h_t j)$ is shown for time steps $t_j = h_t j$, $j = 10^2$, 10^4, 2×10^4, 3×10^4, 5×10^4. We see that the solution $u(x,t)$ develops two moving sharp transition regions, which approach each other and eventually merge into a single one about $x \approx 1/2$. The rational representations of $u(x,t_j)$ have 4, 11, 33, 29, and 19 pairs of conjugate-reciprocal poles, respectively. It also demonstrates that the transition regions of $u(x,t)$ occur within intervals of width of $\mathcal{O}(\nu)$.

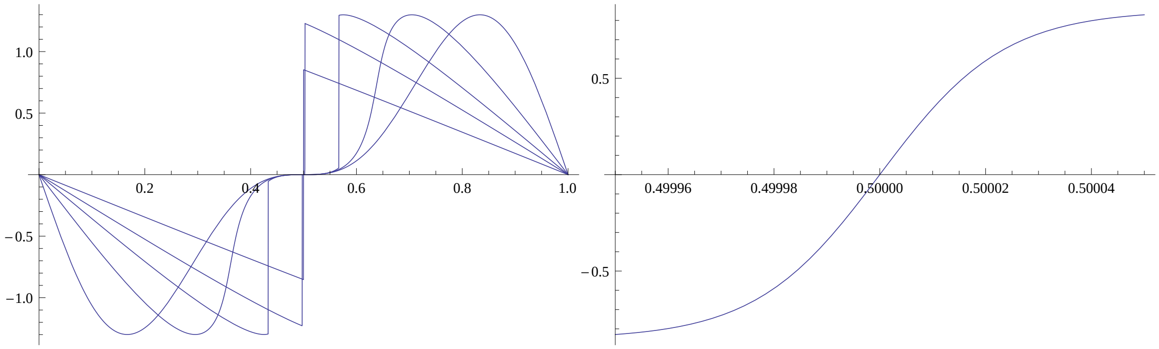


Figure 3.2. (a) Solution $u(x,t_j)$, for $t_j = 10^{-3}$, .1, .2, .3, and .5. (b) $u(x,t_j)$ in the transition region $\left(1/2 - 10^{-5}, 1/2 + 10^{-5}\right)$, for $t_j = 0.4$ (from [27]). These solutions are represented with 4, 11, 33, 29, and 19 pairs of conjugate-reciprocal poles.

4. Accuracy verification

We test the accuracy of Algorithm 4 on 500 random Cauchy matrices, $C_{ij} = \left(\alpha_i\overline{\alpha_j}\right)/\left(1-\gamma_i\overline{\gamma_j}\right)$, $i,j = 1,\ldots,120$. The complex poles $\gamma_j = \rho_j e^{2\pi i\phi_j}$ and residues $\alpha_j = \zeta_j e^{2\pi i\psi_j}$ are generated by taking ρ_j, ϕ_j, and ψ_j from the uniform distribution on $(0,1)$, and taking ζ_j from the uniform distribution on $(0,10)$. For each randomly generated matrix, we first compute, as a gauge, $\overline{C}C = Z\Sigma Z^{-1}$ using the in-built Mathematica^{TM} eigenvalue solver with 300 digits of precision, and compare the result with $\widehat{Z}$ and $\widehat{\Sigma}$ computed via Algorithm 4 using standard double precision. We then evaluate the maximum relative error in the con-eigenvalues $\lambda_j = \Sigma_{jj}$, $\max_j \left|\lambda_j - \widehat{\lambda_j}\right| / |\lambda_j|$, and the maximum error in the computed con-eigenvectors, $\max_j \left\|Z\left(:,j\right) - \widehat{Z}\left(:,j\right)\right\|_2 / \left\|Z\left(:,j\right)\right\|_2$. We first scale $\widehat{Z}\left(:,j\right)$ by the complex-valued constant $Z\left(i_0,j\right)/\widehat{Z}\left(i_0,j\right)$, $i_0 = \max_{1\le i\le n}\left|Z\left(i,j\right)\right|$, since $Z\left(:,j\right)$ and $\widehat{Z}\left(:,j\right)$ are defined only up to an arbitrary complex-valued factor.

Figures 4.1 and 4.2 summarize the result of a typical run. Figure 4.1(a) shows the distribution of the poles γ_j inside the unit disk and Figure 4.1(b) displays $\log_{10}\lambda_j^2$ as a function of the index j. Figures 4.2(a) 4.2(b) show the relative errors in the con-eigenvalues $\left|\lambda_j - \widehat{\lambda_j}\right| / |\lambda_j|$ and the normalized con-eigenvectors $\left\|z_j - \widehat{z_j}\right\|_2 / \left\|z_j\right\|_2$, both as functions of the index j.

In Figures 4.3 and 4.4 for each of the 500 random Cauchy matrices, we plot the error in the computed con-eigenvalues $\left|\widehat{\lambda_j} - \lambda_j\right| / |\lambda_j|$ and con-eigenvectors $\left\|\widehat{z_j} - z_j\right\|_2 / \left\|z_j\right\|_2$for $j = 1, 40, 80, 120$ (note the exponential decay of λ_j). We see that the con-eigenvalues and the con-eigenvectors are computed with nearly full precision for all the Cauchy matrices. In fact, the largest errors $\left|\widehat{\lambda_j} - \lambda_j\right| / |\lambda_j|$ and $\left\|\widehat{z_j} - z_j\right\|_2 / \left\|z_j\right\|_2$ in the computed con-eigenvalues and con-eigenvectors, for any of the 500 Cauchy matrices and any $1 \le j \le n$, are 5.13×10^{-12} and 5.35×10^{-12}.

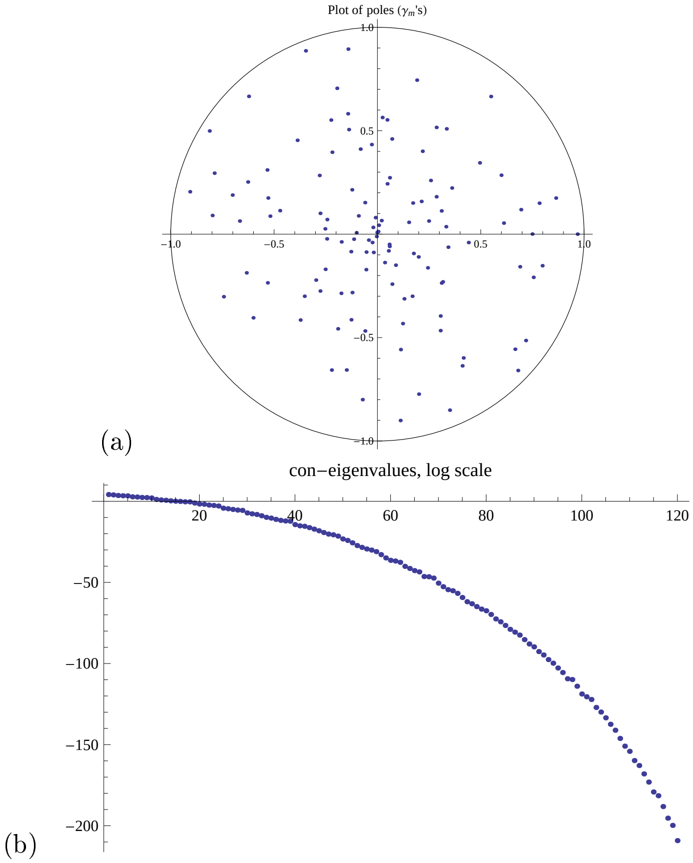


Figure 4.1. (a) Distribution of poles γ_j determining Cauchy matrix C in a typical run. (b) Exponential decay of the eigenvalues λ_j^2 of $\overline{C}C$ as a function of the index j using $\log_{10}$ scale.

5. Accuracy and perturbation bounds

We present error bounds that demonstrate Algorithm 4 of the previous section achieves high relative accuracy. We also provide bounds that demonstrate that small perturbations of a_i, b_j, x_i, and y_j determining C lead to small relative perturbations of the con-eigenvalues and small perturbations of the angles between subspaces spanned by the con-eigenvectors, as long as the parameters x_i and y_j are not too close in a relative sense. In the bounds below, $\|\cdot\|$ denotes the Frobenius norm.

In Theorems 5-7 below we always assume that the con-eigenvalues are simple, although this is not a crucial restriction. In the statements of these theorems, the implicit constant factor implied by the notation $\mathcal{O}(\eta)$ and $\mathcal{O}(\epsilon)$ (here $\epsilon,\eta \ll 1$) depends only on the size n of the matrix C. We note that all these implicit constants may be tracked more carefully and are modest-sized functions of n.

The bounds in the theorems below depend on the Cholesky factors in the decomposition $C = (PL)\,D^2\,(PL)^*$. In particular, the estimates in Theorems 5 - 7 depend on the quantities

$$\begin{aligned}
\mu_0(L) &= \left\|L^{-1}\right\|^2 \kappa(L), \\
\mu_1(L) &= \max\left\{\left\|L^{-1}\right\|^2, \|L\|^2\right\} \kappa(L), \\
\mu_2(L) &= \left\|L^{-1}\right\|^2 \mu_1(L)\,\kappa^3(L),
\end{aligned} \tag{5.1}$$

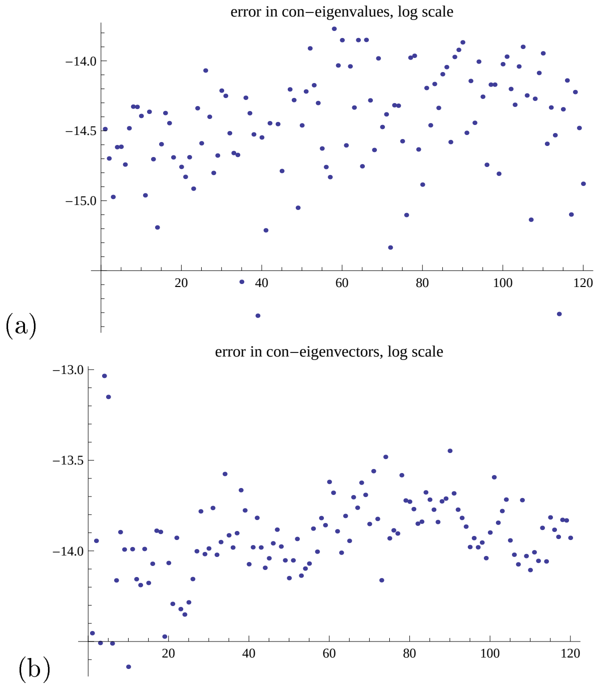


FIGURE 4.2. (a) Relative error in the jth con-eigenvalue, $\left|\lambda_j - \widehat{\lambda_j}\right| / |\lambda_j|$, as a function of the index j. (b) The error in the jth con-eigenvector, $\|z_j - \widehat{z_j}\|_2 / \|z_j\|_2$, $z_j = Z(:, j)$, as a function of the index j.

where the condition number $\kappa(L) = \|L\| \left\|L^{-1}\right\|$ is typically small. The estimates in Theorems 6-7 also depend on

$$\mu_3(L) = \left\|L^{-1}\right\| \left(\rho\mu\psi\mu_2(L) + \nu\kappa^3(L)\right), \tag{5.2}$$

where ρ, μ, and ψ are "pivot growth" factors associated with the QR factorization (see Section 7.3), and the factor ν is associated with the one-sided Jacobi algorithm (see (7.12)).

Remark. There are simple formulas for L_{ij} and $\left(L^{-1}\right)_{ij}$ ([10]) in terms of the parameters a_i, b_j, x_i and y_j defining the Cauchy matrix C, and it is possible that the bounds below may be improved by using this additional structure.

Theorem 5. *Suppose that the parameters defining the positive-definite Cauchy matrix $C = C(a, b, x, y)$ are perturbed to $\tilde{a} = a + \delta a$, $\tilde{b} = b + \delta b$, $x = x + \delta x$, and $y = y + \delta y$. Let us define*

$$\eta = (1/\eta_1 + 1/\eta_2 + 1/\eta_3) \max\left\{\|\delta a\|_\infty, \|\delta b\|_\infty, \|\delta x\|_\infty, \|\delta y\|_\infty\right\},$$

where

$$\eta_1 = \min_{i\neq j} \frac{|x_i - x_j|}{|x_j| + |x_i|}, \quad \eta_2 = \min_{i\neq j} \frac{|y_i - y_j|}{|y_j| + |y_i|}, \quad \eta_3 = \min_{i\neq j} \frac{|x_i + y_j|}{|x_i| + |y_j|}.$$

Let $C = LDL^*$ denote the Cholesky factorization of C, and let $\widetilde{C} = C(\widetilde{a}, \widetilde{b}, \widetilde{x}, \widetilde{y})$ denote the Cauchy matrix corresponding to the perturbed parameters. Finally, let z_i, $\widetilde{z}_i$ denote the con-eigenvectors of C and $\widetilde{C}$, corresponding to con-eigenvalues λ_i and $\widetilde{\lambda}_i$.

Then the relative difference in the con-eigenvalues λ_i and $\widetilde{\lambda}_i$ is bounded as

$$\left|\frac{\lambda_i - \widetilde{\lambda}_i}{\lambda_i}\right| \leq \mu_0(L)\, \mathcal{O}(\eta),$$

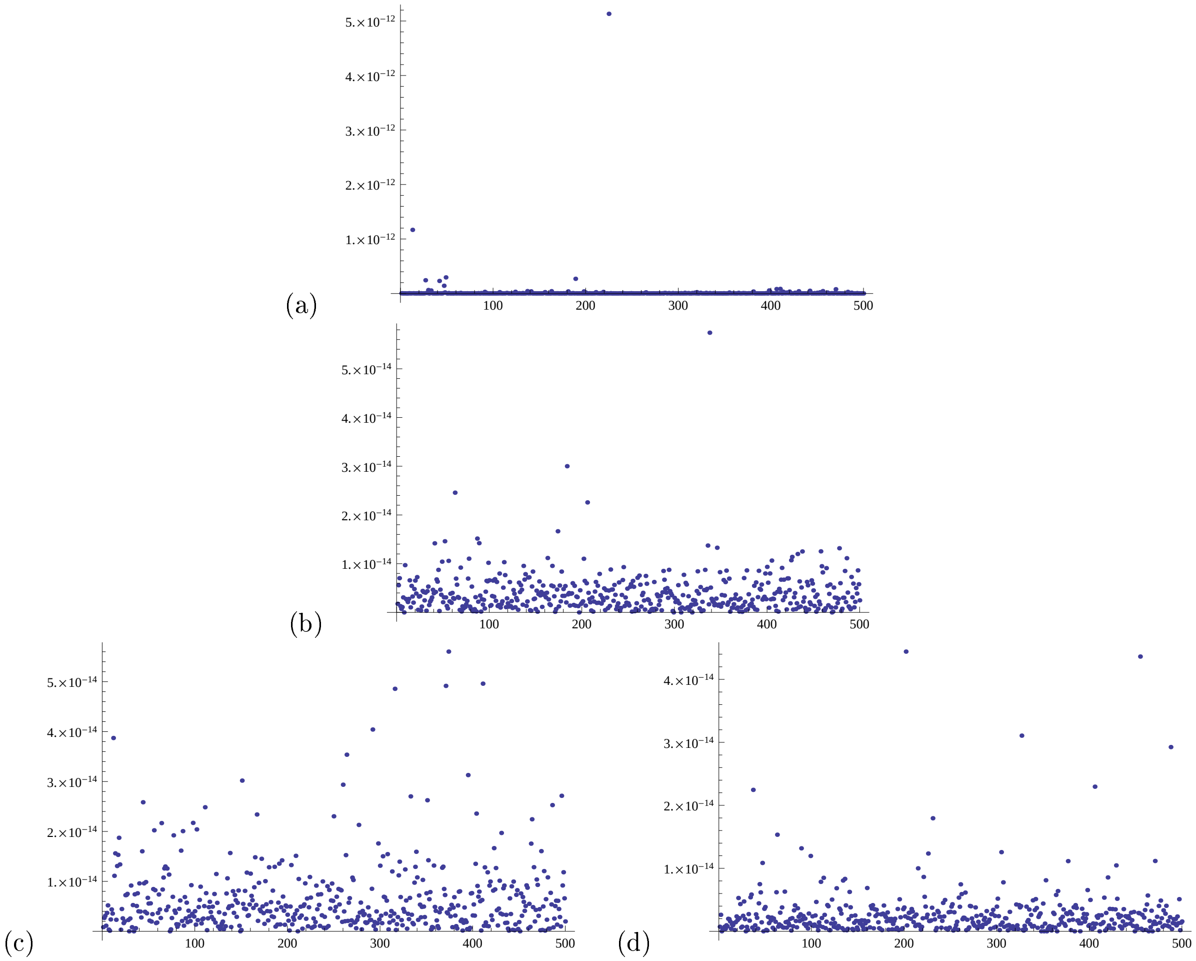


FIGURE 4.3. Relative error in the computed con-eigenvalues, $\left|\widehat{\lambda_j} - \lambda_j\right| / \left|\lambda_j\right|$, for $j = 1, 40, 80, 120$ ((a), (b), (c), and (d), respectively), plotted for each of the 500 random Cauchy matrices.

and the acute angle between the con-eigenvectors z_i and $\widetilde{z}_i$ is bounded by

$$\sin\left(\angle z_i, \widetilde{z}_i\right) \leq \kappa\left(L\right)\left(\frac{\mu_2\left(L\right)}{relgap_i} + \mu_0\left(L\right)\mu_1\left(L\right)\right)\mathcal{O}\left(\eta\right).$$

Here $\mu_0\left(L\right)$, $\mu_1\left(L\right)$ and $\mu_2\left(L\right)$ are defined in (5.1), and

$$relgap_i = \min_{j \neq i} \frac{\left|\lambda_i - \lambda_j\right|}{\left|\lambda_i\right| + \left|\lambda_j\right|}.$$

Next we state

Theorem 6. *Suppose that Algorithm 4 is used to compute the full con-eigenvalue decomposition of a positive-definite Cauchy matrix C. Suppose also that C has the Cholesky factorization $C = \left(PL\right)D^2\left(PL\right)^*$, where P is the permutation matrix that encodes complete pivoting.*

Then the relative error between the computed con-eigenvalue $\widehat{\lambda_i}$ and the exact λ_i is bounded as

$$\frac{\left|\widehat{\lambda_i} - \lambda_i\right|}{\left|\lambda_i\right|} \leq \left(\rho\mu\psi\mu_0\left(L\right) + \nu\right)\mathcal{O}\left(\epsilon\right),$$

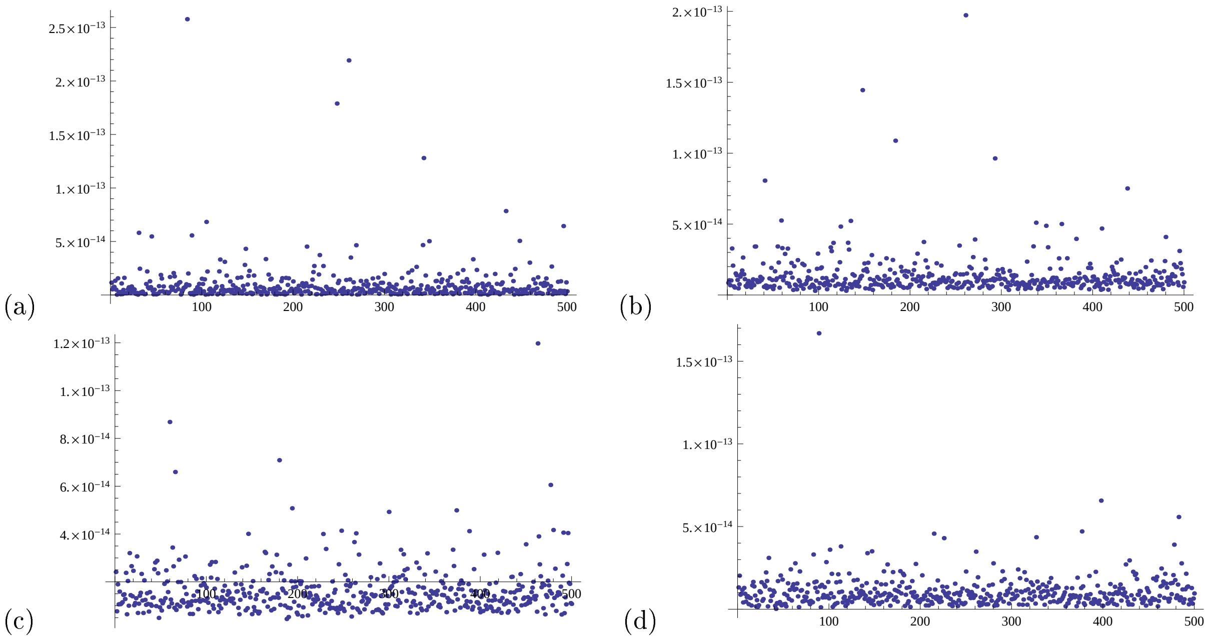


Figure 4.4. Relative error in the computed con-eigenvectors, $\|\widehat{z}_j - z_j\|_2/\|z_j\|_2$, for $j = 1, 40, 80, 120$ ((a), (b), (c), and (d), respectively), plotted for each of the 500 random Cauchy matrices.

where ρ, μ, and ψ are "pivot growth" factors associated with the QR factorization (see Section 7.3), and the factor ν is associated with the one-sided Jacobi algorithm (see (7.12)).

Letting z_i, $\widehat{z}_i$ denote exact and computed con-eigenvectors of C, the acute angle between z_i and $\widehat{z}_i$ then satisfies

$$\sin\left(\angle\widehat{z}_i, z_i\right) \leq \kappa\left(L\right)\left(\frac{\mu_3\left(L\right)}{relgap_i} + \left\|L^{-1}\right\|^2\kappa^3\left(L\right)\right)\mathcal{O}\left(\epsilon\right),$$

where $relgap_i$ is defined as in Theorem 5 and $\mu_3\left(L\right)$ is defined in 5.2.

Theorem 7. *Suppose Algorithm 4 is used to compute m approximate con-eigenvalues and con-eigenvectors of a positive-definite Cauchy matrix C. Suppose also that C has the Cholesky factorization $C = \left(PL\right)D^2\left(PL\right)^*$, where P is the permutation matrix that encodes complete pivoting. Assuming that $D^2_{mm} \leq \lambda_i\epsilon$ for some $1 \leq i \leq m$, the following error bound holds for the computed con-eigenvalue $\widehat{\lambda}_i$,*

$$\frac{\left|\widehat{\lambda}_i - \lambda_i\right|}{\left|\lambda_i\right|} \leq \left(\rho\mu\psi\mu_0\left(L\right) + \nu + \left\|C\right\|\mu_1^2\left(L\right)\right)\mathcal{O}\left(\epsilon\right),$$

and the acute angle between z_i and $\widehat{z}_i$ is bounded by

$$\sin\left(\angle\widehat{z}_i, z_i\right) \leq \kappa\left(L\right)\left(\frac{\mu_3\left(L\right) + \left\|C\right\|\mu_1^2\left(L\right)}{relgap_i} + \left\|L^{-1}\right\|^2\kappa^3\left(L\right)\right)\mathcal{O}\left(\epsilon\right).$$

In the above estimates, ρ, μ, and ψ are "pivot growth" factors associated with the QR factorization (see Section 7.3), and the factor ν is associated with the one-sided Jacobi algorithm (see (7.12)).

The proofs of the theorems in this section may be found in the online version of this paper [28].

Remark 8. We note that the constants in the theorems above are significantly more pessimistic than we actually observe in numerical experiments. Indeed, while the bounds on the con-eigenvectors depend only on the well-conditioned matrix L (and, in particular, are independent of the exponentially decaying diagonal matrix D), they still scale like $\kappa^9(L)$; the bounds on the con-eigenvalues are better—they scale like $\kappa^3(L)$. However, in practice Algorithm 4 achieves nearly full precision for all the con-eigenvalues and con-eigenvectors. While it is likely that better estimates can be obtained, those presented here elucidate the basic mechanism behind the high accuracy that we observe in our experiments.

6. Discussion: comparison with related approaches for constructing optimal rational approximations

Numerical approaches for finding near optimal rational approximations originate in theoretical results of Adamyan, Arov, and Krein [1, 2, 3]. In particular, given a periodic function $f\left(e^{2\pi ix}\right) \in L^{\infty}(0,1)$, AAK theory yields an optimal "rational-like" approximation $r_M\left(e^{2\pi ix}\right)$,

$$r_M(z) = \frac{a_0 + a_1 z + a_2 z^2 + \dots}{(z-\zeta_1)\dots(z-\zeta_M)}, \quad |\zeta_j| < 1, \tag{6.1}$$

constructed from the left and right singular vectors corresponding to the Mth singular value, σ_M, of the infinite Hankel matrix $H_{ij} = \hat{f}(i+j-1)$, $i,j = 1,2,\dots$. The numerator of $r_M(z)$ in (6.1) is analytic in the unit disk. The approximation error satisfies

$$\max_x \left| f\left(e^{2\pi ix}\right) - r_M\left(e^{2\pi ix}\right) \right| = \sigma_M,$$

where the number of poles ζ_j in (6.1) equals the index M of the singular value σ_M (index counting starts from zero). Moreover, the L^{∞}-norm approximation error is optimal among all functions of the form (6.1).

In order to use AAK theory to compute (near) optimal rational approximations, standard numerical approaches compute singular vectors of a truncated Hankel matrix. The poles of the rational approximation are obtained as roots of a polynomial whose coefficients are the entries of the singular vector. Such approaches have a long history of their own and, in particular, let us mention the pioneering papers [37, 38, 39]. A recent version (incorporating additional ideas) can be found in [22].

Instead of truncating the Hankel matrix, the approach of this paper is based on the observation that it is always possible (see e.g. [6, 8, 5, 11]) to construct a sub-optimal rational approximation, i.e., an approximation with excessive number of poles for a desired accuracy. This leads us to specialize AAK theory to proper rational functions $f\left(e^{2\pi ix}\right)$, and to formulate the reduction problem (see Section 2.1 and [6, Section 6]). Importantly, this results in a con-eigenvalue problem of finite size and with no additional approximations. Moreover, this formulation allows us to develop a numerical calculus based on rational functions (numerical operations such as addition and multiplication increase the number of poles; the reduction algorithm is applied to keep their number near optimally small, see [27]). Early approaches of this type can be found in [32, 14, 40]; however, these algorithms may require extended precision for high accuracy and also scale cubically in the number of original poles.

Comparing our approach with that in e.g. [22], we make two observations. First, to justify the truncation of an infinite Hankel matrix, the Fourier coefficients have to decay below the desired accuracy of approximation. Thus, for functions that have sharp transitions (as in the example of Section 3.2) or singularities (as in the example of Section 3.1), where the Fourier coefficients decay slowly, this would require computing singular values of very large matrices. In the examples of Sections 3.1 and 3.2, Hankel matrices of size $\approx 10^7 \times 10^7$ and $\approx 10^6 \times 10^6$ would be needed in order to attain a comparable accuracy. This approach would also require finding roots of polynomials with $\approx 10^7$ and $\approx 10^6$ coefficients, respectively.

Our second observation is that using Hankel matrices may require extended precision arithmetic if high accuracy is desired, as is the case in examples of Sections 3.1 and 3.2. Indeed, existing SVD algorithms do not accurately compute small singular values of Hankel matrices. Also, the roots of high degree polynomials (determined at the SVD step) may be sensitive to perturbations in their coefficients. However, when limited to approximating smooth functions, these “truncated Hankel” methods can yield surprisingly high accuracy since the errors in the poles may be compensated by the residues. As far as we are aware, truncated Hankel methods for constructing optimal rational approximations for functions with singularities generally do not achieve approximation errors better than $\approx 10^{-4}$. In contrast, in Section 3.1 we show that the reduction algorithm approximates piecewise smooth functions with errors close to machine precision.

We also note that the results in [27] (illustrated in Section 3.2) demonstrate an effective numerical calculus based on the reduction algorithm, capable of computing highly accurate solutions to viscous Burgers' equation for viscosity as small as 10^{-5}. These solutions exhibit moving transitions regions of width $\approx 10^{-5}$, and computing them with high accuracy over long time intervals is a nontrivial task for any numerical method. The con-eigenvalue algorithm of this paper is critical to the high accuracy and efficiency of this numerical calculus.

7. Appendix: background on algorithms for high relative accuracy

Here we provide necessary background on computing highly accurate SVDs. Although the results we need in [20, 33, 17, 34, 15, 29] are only stated there for real-valued matrices, they carry over to complex-valued matrices with minor modifications and are formulated as such.

7.1. **Accurate SVDs of matrices with rank-revealing decompositions.**

According to the usual perturbation theory for the SVD (see e.g. [12]), perturbations δA of a matrix A change the ith singular value σ_i by $\delta\sigma_i$ and corresponding unit eigenvector u_i by δu_i, where (assuming for simplicity that σ_i is simple),

$$\left|\delta\sigma_i\right|/\sigma_1 \leq \left\|\delta A\right\|, \quad \left\|\delta u_i\right\| \leq \frac{\left\|\delta A\right\|}{\mathrm{absgap}_i}, \quad \mathrm{absgap}_i = \min_{i\neq j}\left|\sigma_i-\sigma_j\right|/\sigma_1. \tag{7.1}$$

Therefore, small perturbations in the elements of A may lead to large relative changes in the small singular values and the associated singular vectors. Moreover, since standard algorithms compute an SVD of some nearby matrix $A+\delta A$, where $\left\|\delta A\right\|/\left\|A\right\| = \mathcal{O}\left(\epsilon\right)$, the perturbation bound (7.1) shows that the computed small singular values and corresponding singular vectors will be inaccurate.

In contrast, the authors in [17] show that, for many structured matrices, the ith singular value $\sigma_i \ll \sigma_1$ and the associated singular vector are robust with respect to small perturbations of the matrix that preserve its underlying structure. The sensitivity is instead governed by the ith *relative* gap

$$\mathrm{relgap}_i = \min_{i\neq j}\frac{\left|\sigma_i-\sigma_j\right|}{\sigma_i+\sigma_j}.$$

More precisely, let us consider the class of matrices for which a rank-revealing decomposition $A = XDY^*$ is available and may be computed accurately. Here X and Y are $n\times m$ well-conditioned matrices and D is an $m\times m$ diagonal matrix that contains any possible ill-conditioning of A. As is shown in [17], a perturbation of $A = XDY^*$ that is of the form $A+\delta A = \left(X+\delta X\right)\left(D+\delta D\right)\left(Y+\delta Y\right)^*$, where

$$\frac{\left\|\delta X\right\|}{\left\|X\right\|} = \mathcal{O}\left(\epsilon\right), \quad \frac{\left\|\delta Y\right\|}{\left\|Y\right\|} = \mathcal{O}\left(\epsilon\right), \quad \frac{\left|\delta D_{ii}\right|}{\left|D_{ii}\right|} = \mathcal{O}\left(\epsilon\right), \tag{7.2}$$

changes the ith singular value σ_i and associated left (or right) singular vector u_i by amounts $\delta\sigma_i$ and δu_i bounded by

$$\frac{\left|\delta\sigma_i\right|}{\sigma_i} \leq \max\left(\kappa\left(X\right),\kappa\left(Y\right)\right)\mathcal{O}\left(\epsilon\right), \quad \left\|\delta u_i\right\| \leq \frac{\max\left(\kappa\left(X\right),\kappa\left(Y\right)\right)}{\mathrm{relgap}_i}\mathcal{O}\left(\epsilon\right), \tag{7.3}$$

where $\kappa(X) = \left\|X\right\|\left\|X^{\dagger}\right\|$ and $X^{\dagger}$ denotes the pseudo-inverse of A. One reason this class of matrices is so useful is that Gaussian elimination with complete pivoting (GECP) (or simple modifications) computes accurate rank-revealing decompositions of many types of structured matrices (see [17] and

[15]). Moreover, small perturbations of such matrices that preserve their underlying structure lead to small perturbations in the rank-revealing factors and, therefore, small relative perturbations of the singular values.

Given the decomposition $A = XDY^*$, it is shown in [17, Algorithm 3.1] that an SVD of A may be computed with high relative accuracy, and with about the same cost as standard, less accurate SVD algorithms for dense matrices. The key to this algorithm is the one-sided Jacobi algorithm (briefly reviewed in Section 7.4), which, with an appropriate stopping criterion, accurately computes the SVD of matrices of the form DB, where D is diagonal (and typically highly ill-conditioned) and B is well-conditioned (see [20] and [33]). In particular, the algorithm in [17, Algorithm 3.1] yields computed singular values $\widehat{\sigma_i}$ and left (or right) singular vectors $\widehat{u_i}$ that satisfy

$$\frac{|\sigma_i - \widehat{\sigma_i}|}{\sigma_i} \leq \max\left(\kappa\left(X\right), \kappa\left(Y\right)\right) \mathcal{O}\left(\epsilon\right), \tag{7.4}$$

$$\|u_i - \widehat{u_i}\| \leq \frac{\max\left(\kappa\left(X\right), \kappa\left(Y\right)\right)}{\mathrm{relgap}_i} \mathcal{O}\left(\epsilon\right), \tag{7.5}$$

7.2. **LDU factorization of Cauchy matrices.** In this section we review how a modification of GECP computes accurate rank-revealing decompositions of Cauchy matrices [15].

We describe Demmel's algorithm (see Algorithms 3 and 4 in [15] and Algorithm 2.5 in [9]) for computing an accurate rank-revealing decomposition of a $n \times n$ positive-definite Cauchy matrix $C_{ij} = a_i b_j / \left(x_i + y_j\right)$ (note that Demmel refers to such matrices as quasi-Cauchy). The algorithm is based on a modification of Gaussian elimination for computing, in $\mathcal{O}\left(n^2\right)$ operations, the Cholesky factorization $C = \left(PL\right) D \left(PD\right)^*$ of a positive-definite Cauchy matrix (more generally, the algorithm computes an LDU factorization for an arbitrary Cauchy matrix in $\mathcal{O}\left(n^3\right)$ operations). Here P is a permutation matrix, L is a unit lower triangular matrix, and D is a diagonal matrix with positive diagonal elements. It is shown in [15] that, remarkably, the components of the LDU factors $\widehat{L}$, $\widehat{U}$, and $\widehat{D}$ are computed to high relative accuracy,

$$\left|\widehat{L}_{ij} - L_{ij}\right| \leq \left|L_{ij}\right| c_n \epsilon, \quad \left|\widehat{U}_{ij} - U_{ij}\right| \leq c_n \left|U_{ij}\right| \epsilon, \quad \left|\widehat{D}_{ii} - D_{ii}\right| \leq c_n \left|D_{ii}\right| \epsilon, \tag{7.6}$$

where c_n is a modest-sized function of n. The basic reason the algorithm achieves high relative accuracy is that the only operations involved are multiplication and division of floating point numbers (additions and subtractions in the algorithm involve only x_i and y_j, which are assumed to be exact).

We now review the basic idea behind the algorithm in [15]. First, ignoring pivoting for a moment, we assume that, after k steps of Gaussian elimination, the Cauchy matrix is transformed to the matrix $G^{(k)}$,

$$G^{(k)} = \left(\begin{array}{cc} G_{11}^{(k)} & G_{12}^{(k)} \\ 0 & G_{22}^{(k)} \end{array}\right).$$

The elements of the Schur complement $G_{22}^{(k+1)}$ may be computed from those of $G_{22}^{(k)}$ by using the recursion

$$G_{ij}^{(k)} \;=\; \left(\frac{x_i - x_k}{x_i + y_k}\right)\left(\frac{y_j - y_k}{y_j + x_k}\right) G_{ij}^{(k-1)}, \quad i,j = k+1, \ldots, n. \tag{7.7}$$

Introducing pivoting, we observe that the matrix $G^{(k)}$ may be obtained by applying Gaussian elimination to a Cauchy matrix $C^{(k)} = C^{(k)}\left(a^{(k)}, b^{(k)}, x^{(k)}, y^{(k)}\right)$, where $a^{(k)}$, $b^{(k)}$, $x^{(k)}$ and $y^{(k)}$ are permutations of a, b, x and y corresponding to the row and column pivoting of C. As long as the vectors a, b, x and y are permuted according to the pivoting of $G^{(k)}$, the recursive formula (7.7) still holds.

It is observed in [15] that if C is positive-definite (and, therefore, only diagonal pivoting is needed), then the pivot order may be determined in advance in $\mathcal{O}\left(n^2\right)$ operations by computing $\mathrm{diag}\left(G^{(k)}\right)$ from formula (7.7). Once the correct pivot order is known, we do not need to compute the entire

Schur complement $G^{(k)}$ to extract the components of L and U, but only its kth row and kth column. Indeed, we may use Algorithm 2.5 in [9], which uses the displacement structure of C, to compute an accurate Cholesky decomposition in $\mathcal{O}\left(n^2\right)$ operations. To see how, note that it easily follows from (7.7) that the Schur complement of a Cauchy matrix is a Cauchy matrix,

$$G^{(k)}\left(i,j\right)=\frac{\alpha_i^{(k)}\beta_j^{(k)}}{x_i+y_j},\quad i,j=k+1,\ldots,n, \tag{7.8}$$

where the parameters $\alpha_i^{(k)}$ and $\beta_i^{(k)}$ satisfy the recursion

$$\alpha_i^{(k)}=\frac{x_i-x_k}{x_i+y_k}\alpha_i^{(k-1)},\quad \beta_i^{(k)}=\frac{y_i-y_k}{y_i+x_k}\beta_i^{(k-1)},\quad i=k+1,\ldots,n. \tag{7.9}$$

Since the kth column $L\left(:,k\right)$ may be extracted from $G^{(k)}\left(:,k\right)$, we therefore only require $\mathcal{O}\left(n\right)$ operations at each step of Gaussian elimination to compute $L\left(:,k\right)$. Updating $\alpha_i^{(k)}$ and $\beta_i^{(k)}$ also requires only $\mathcal{O}\left(n\right)$ operations. In Section 2.3 (see Algorithms 2 and 3), we present an $\mathcal{O}\left(n\left(\log\delta^{-1}\right)^2\right)$ algorithm to compute con-eigenvalues greater than a user specified cutoff δ and, as a result, yielding a fast algorithm for obtaining nearly optimal rational approximations. Once an accurate LDU factorization $C\approx\left(P\widehat{L}\right)\widehat{D}\left(P\widehat{D}\right)^*$ is available, an accurate SVD of C may be obtained using the algorithm in [17, Algorithm 3.1].

7.3. **Rank-revealing decompositions of graded matrices.** We also review how a variant of the QR Householder algorithm with complete pivoting computes accurate rank-revealing decompositions of graded matrices [29].

It is shown in [29] that the Householder QR algorithm with complete pivoting may be used to compute a rank-revealing decomposition of a graded matrix of the form $A=D_1BD_2$. Here D_1 and D_2 are diagonal matrices that account for the ill-conditioning of A. Recall that the Householder QR algorithm uses repeated applications of orthogonal matrices to reduce A to an upper-triangular matrix R. On the first step, the parameter β_1 and the vector v_1 of the Householder reflection matrix $Q^{(1)}=I-\beta_1v_1v_1^*$ are chosen so that

$$Q^{(1)}\begin{pmatrix}a_{11}\\a_{21}\\\vdots\\a_{n1}\end{pmatrix}=\begin{pmatrix}a_{11}^{(1)}\\0\\\vdots\\0\end{pmatrix}.$$

Consequently, the first application of $Q^{(1)}$ to A results in a matrix of the form

$$A^{(1)}=Q^{(1)}A=\begin{pmatrix}a_{11}^{(1)}&a_{12}^{(1)}&\ldots&a_{1n}^{(1)}\\0&a_{22}^{(1)}&\ldots&a_{2n}^{(1)}\\\vdots&\vdots&\ddots&\vdots\\0&a_{n2}^{(1)}&\ldots&a_{nn}^{(1)}\end{pmatrix}.$$

This process is repeated on the $(n-1)\times(n-1)$ lower block $\left[a_{ij}^{(1)}\right]_{2\leq i,j\leq n}$ and, after $n-1$ such steps, $A^{(n-1)}=Q^{(n-1)}\ldots Q^{(1)}A=R$, where R is upper triangular. In the version considered in [29], the rows of A are first pre-sorted so that so that $\left\|A\left(1,:\right)\right\|_\infty\geq\cdots\geq\left\|A\left(n,:\right)\right\|_\infty$. The algorithm then proceeds as above, except that at each step, k, column pivoting is performed to ensure that $\left\|A^{(k)}\left(k:n,k\right)\right\|_2\geq\cdots\geq\left\|A^{(k)}\left(k:n,n\right)\right\|_2$. Letting P_1 denote the row permutation matrix that pre-sorts the rows of A, and letting P_2 denote the column permutation matrix corresponding to the column pivoting, the QR Householder algorithm produces the QR factorization $P_1AP_2=QR$.

Following [29], we consider the error analysis of the Householder algorithm (without pivoting) applied to P_1AP_2, where P_1 and P_2 are chosen so that no column or row exchanges are necessary (e.g.

the matrix A is pre-pivoted). Assume that the matrix P_1AP_2 may be factored as $P_1AP_2 = D_1BD_2$, where D_1 and D_2 are diagonal matrices, and that the Householder algorithm, applied to the row-scaled matrix $C = D_1B$, produces intermediate matrices $C^{(k)}$ with columns $c_j^{(k)}$. Finally, define the quantities ρ, μ, and ψ by

$$\rho = \max_i \frac{\max_{j,k}\left|c_{ij}^{(k)}\right|}{\max_j |c_{ij}|}, \quad \mu = \max_k \max_{j\geq k} \frac{\left\|c_j^{(k)}\left(k:m\right)\right\|}{\left\|c_k^{(k)}\left(k:m\right)\right\|}, \quad \psi = \max_{\substack{1\leq i\leq n\\ i\leq k\leq n}} \frac{\max_j |c_{kj}|}{\max_j |c_{ij}|}. \tag{7.10}$$

The above quantities measure the extent to which the Householder algorithm preserves the scaling in the intermediate matrices $A^{(k)}$, and are almost always small (this is analogous to the pivot growth factor in Gaussian elimination with row pivoting). It is shown in [29] that

Theorem 9. *Suppose that A is pre-pivoted, and the Householder algorithm is used to compute the upper triangular matrix $\widehat{R}$ of the QR decomposition. Then there is an orthogonal matrix Q such that $Q\widehat{R} = D_1\left(B + \delta B\right)D_2$, where δB satisfies*

$$\|\delta B\| \leq \rho\psi\mu \|B\| \mathcal{O}\left(\epsilon\right),$$

and ρ, μ, and ψ are defined in (7.10).

In [29] Theorem 9 is combined with the theory developed in [17] (e.g., see Theorems 4.1 and 4.2 in [17]) to show that the QR algorithm with complete pivoting produces accurate rank revealing decompositions of graded matrices of the form $A = D_1BD_2$, as long as the principal minors of B are well-conditioned and the diagonal elements of D_1 and D_2 are approximately decreasing in magnitude.

Remark. Instead of pre-sorting the rows of A and applying the Householder algorithm with column pivoting, one may also use a version of the Householder algorithm in which both row and column pivoting is employed (see [29] for more details). Gaussian elimination with complete pivoting may also be used to obtain accurate rank-revealing decompositions of graded matrices [17].

7.4. **Modified one-sided Jacobi algorithm .** The heart of the algorithm in [17, Algorithm 3.1] is the modified one-sided Jacobi algorithm, which accurately computes the SVD of matrices of the form DB and BD, where D is diagonal and typically highly graded, and B is well-conditioned (see [20], [33], [24, 25]). Although we focus on the one-sided Jacobi algorithm as applied to $G = BD$, analogous considerations apply to $G = DB$ by replacing G by G^*. The one-sided Jacobi algorithm works by applying a sequence of Jacobi matrices $J_1, \ldots, J_M$ to G from the right (i.e., the same side as the scaling, which ensures that components of the right singular vectors are computed with high relative accuracy). Each Jacobi matrix J is chosen to orthogonalize two selected columns, and one sweep consists of orthogonalizing columns in the order $(1,1),(1,2),\ldots,(1,n)$, followed by columns $(2,3),(2,4),\ldots,(2,n)$, and so on. Sweeps are repeated until all the columns are orthogonal to each other to within the bound

$$G\left(J_1\cdots J_M\right) = W, \quad \frac{|w_i^*w_j|}{|w_i^*w_i|^{1/2}\,|w_i^*w_i|^{1/2}} \leq n\epsilon, \text{ if } i\neq j.$$

This stopping criterion is used to ensure that even the smallest singular values are computed with high relative accuracy. The SVD of $G = U\Sigma V^*$ immediately follows by taking $\Sigma_{ii} = W\left(:,i\right)$, $V = W/\Sigma$, and $U = \left(J_1J_2\cdots J_M\right)^*$.

It will be crucial for the error bounds developed in this paper that the components of the left singular vectors of DB (or the right singular vectors of BD) scale in a way similar to D, and are computed accurately relative to this scaling. At each step m of the Jacobi algorithm, we write $\left(J_0\cdots J_m\right)G = B_mD_m$, where the columns of B_m have unit l^2-norm and the matrix D_m is diagonal. We also define

$$\nu_0 = \max_{1\leq m\leq M} \kappa_2\left(B_m\right), \tag{7.11}$$

and

$$\nu=\rho\left(M,n\right)\nu_{0}^{2}, \tag{7.12}$$

where $\rho\left(M,n\right)$ is proportional to $M\cdot n^{3/2}$, and ν_{0} in defined in (7.11). Then we have the following result from [33] and [20].

Theorem 10. *Let $G=DB$ be a $n\times n$ full-rank, complex-valued matrix, where the diagonal matrix D is chosen so that the l^{2}-norm of each column of B is unity. Suppose that one-sided Jacobi algorithm is used to compute an approximation $\widehat{u_{i}}$ to the ith left singular vector u_{i} of G, corresponding to singular value Σ_{ii}, and the iteration converges after M sweeps. Then the following error bound holds on the computed components of u_{i}:*

$$\left|u_{i}\left(j\right)-\widehat{u_{i}}\left(j\right)\right|\leq\min\left\{ \frac{D_{jj}}{\sqrt{\Sigma_{ii}}},\frac{\sqrt{\Sigma_{ii}}}{D_{jj}}\right\} \left(\frac{\nu}{relgap_{i}}\epsilon+\mathcal{O}\left(\epsilon^{2}\right)\right), \tag{7.13}$$

where

$$\mathrm{relgap}_{i}=\frac{\left|\sigma_{i}-\sigma_{j}\right|}{\sigma_{i}+\sigma_{j}}.$$

Moreover, the computed singular value $\widetilde{\Sigma_{ii}}$ satisfies

$$\frac{\left|\Sigma_{ii}-\widetilde{\Sigma_{ii}}\right|}{\Sigma_{ii}}\leq\nu_{0}\mathcal{O}\left(\epsilon\right).$$

Acknowledgement

We would like to thank James Demmel for providing Matlab code for computing the SVD of Cauchy matrices. We also would like to thank the anonymous referees whose comments helped to improve the paper.

References

[1] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Infinite Hankel matrices and generalized Carathéodory-Fejér and I. Schur problems. *Funkcional. Anal. i Priložen.*, 2(4):1–17, 1968.

[2] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Infinite Hankel matrices and generalized problems of Carathéodory-Fejér and F. Riesz. *Funkcional. Anal. i Priložen.*, 2(1):1–19, 1968.

[3] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Analytic properties of the Schmidt pairs of a Hankel operator and the generalized Schur-Takagi problem. *Mat. Sb. (N.S.)*, 86(128):34–75, 1971.

[4] J. Barlow and J. Demmel. Computing accurate eigensystems of scaled diagonally dominant matrices. Technical report, New York University, Technical Report 421, 1988.

[5] G. Beylkin, R.D. Lewis, and L. Monzón. On the Design of Highly Accurate and Efficient IIR and FIR Filters. *IEEE Trans. Signal Process.*, 2011. http://dx.doi.org/10.1109/TSP.2012.2197397.

[6] G. Beylkin and L. Monzón. On approximation of functions by exponential sums. *Appl. Comput. Harmon. Anal.*, 19(1):17–48, 2005.

[7] G. Beylkin and L. Monzón. Nonlinear inversion of a band-limited Fourier transform. *Appl. Comput. Harmon. Anal.*, 27(3):351–366, 2009.

[8] G. Beylkin and L. Monzón. Approximation of functions by exponential sums revisited. *Appl. Comput. Harmon. Anal.*, 28(2):131–149, 2010.

[9] T. Boros, T. Kailath, and V. Olshevsky. Pivoting and backward stability of fast algorithms for solving Cauchy linear equations. *Linear Algebra Appl.*, 343/344:63–99, 2002. Special issue on structured and infinite systems of linear equations.

[10] Choong Yun Cho. On the triangular decomposition of Cauchy matrices. *Math. Comp.*, 22:819–825, 1968.

[11] A. Damle, G. Beylkin, T. S. Haut, and L. Monzón. Near optimal rational approximations of large data sets. *Appl. Comput. Harmon. Anal.*, 2012. Submitted.

[12] C. Davis and W. M. Kahan. Some new bounds on perturbation of subspaces. *Bulletin of the American Mathematical Society*, 75(4):863–&, 1969.

[13] P. Deift, J. Demmel, C. Li, and C. Tomei. The bidiagonal singular value decomposition and hamiltonian mechanics. *SIAM J. Num. Anal*, 28:1463–1516, 1991.

[14] Ph. Delsarte, Y. Genin, and Y. Kamp. On the role of the Nevanlinna-Pick problem in circuit and system theory. *International Journal of Circuit Theory and Applications*, 9(2):177–187, 1981.

[15] J. Demmel. Accurate singular value decompositions of structured matrices. *SIAM J. Matrix Anal. Appl.*, 21(2):562–580, 1999.

[16] J. Demmel, I. Dumitriu, O. Holtz, and P. Koev. Accurate and efficient expression evaluation and linear algebra. *Acta Numer.*, 17:87–145, 2008.

[17] J. Demmel, M. Gu, S. Eisenstat, I. Slapnicar, K. Veselic, and Z. Drmac. Computing the singular value decomposition with high relative accuracy. *LAPACK Working Note*, 119(CS-97-348), 1997.

[18] J. Demmel, M. Gu, S. Eisenstat, I. Slapnivcar, K. Veselic, and Z. Drmac. Computing the singular value decomposition with high relative accuracy. *Linear Algebra Appl.*, 299(1-3):21–80, 1999.

[19] J. Demmel and W. Kahan. Accurate singular values of bidiagonal matrices. *SIAM J. Sci. Stat. Comput*, 11:873–912, 1990.

[20] J. Demmel and K. Veselic. Jacobi's method is more accurate than QR. *SIAM. J. Matrix Anal. and Appl.*, 13(4):1204–1245, 1992.

[21] J. W. Demmel and W. Gragg. On computing accurate singular values and eigenvalues of matrices with acyclic graphs. *Linear Algebra Appl.*, 185:203–217, 1993.

[22] J. V. Deun and L. N. Trefethen. A robust implementation of the Carathéodory-Fejér method for rational approximation. *BIT Numerical Mathematics*, vol. 51(No. 4), 2011.

[23] Z. Drmač. Accurate computation of the product-induced singular value decomposition with applications. *SIAM Journal of Numerical Analysis*, 35(5):1969–1994, 1998.

[24] Z. Drmac and K. Veselic. New fast and accurate Jacobi SVD algorithm. I. *SIAM J. Matrix Anal. Appl.*, 29(4):1322–1342, 2007.

[25] Z. Drmac and K. Veselic. New fast and accurate Jacobi SVD algorithm. II. *SIAM J. Matrix Anal. Appl.*, 29(4):1343–1362, 2007.

[26] K. V. Fernando and B. N. Parlett. Accurate singular values and differential qd algorithms. *Numer. Math.*, 67(2):191–229, 1994.

[27] T. Haut, G. Beylkin, and L. Monzón. Solving Burgers' equation using optimal rational approximations. *Appl. Comput. Harmon. Anal.*, 2012. http://dx.doi.org/10.1016/j.acha.2012.03.004 (electronic).

[28] T. S. Haut and G. Beylkin. Fast and accurate con-eigenvalue algorithm for optimal rational approximations. *arXiv:1012.3196 [math.NA]*, 2011.

[29] N. J. Higham. *QR* factorization with complete pivoting and accurate computation of the SVD. In *Proceedings of the International Workshop on Accurate Solution of Eigenvalue Problems (University Park, PA, 1998)*, volume 309, pages 153–174, 2000.

[30] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 1990.

[31] P. Koev. Accurate eigenvalues and SVDs of totally nonnegative matrices. *SIAM J. Matrix Anal. Appl*, 27:1–23, 2005.

[32] S.Y. Kung. Optimal Hankel-norm model reductions: Scalar systems. In *Proceedings of the Joint Automatic Control Conference*, number FA8D, 1980.

[33] R. Mathias. Accurate eigensystem computations by Jacobi methods. *SIAM J. Matrix Anal. Appl*, 16:977–1003, 1996.

[34] R. Mathias. Spectral perturbation bounds for positive definite matrices. *SIAM J. Matrix Anal. Appl*, 18:959–80, 1997.

[35] D. J. Newman. Rational approximation of |x|. *Michigan Math. J.*, 11:11–14, 1964.

[36] I. Slapnicar. Highly accurate symmetric eigenvalue decomposition and hyperbolic SVD. *Linear Algebra and its Applications*, 358:387–424, 2003.

[37] L. N. Trefethen. Rational Chebyshev approximation on the unit disk. *Numer. Math.*, 37(2):297–320, 1981.

[38] L. N. Trefethen and M. H. Gutknecht. The Carathéodory-Fejér method for real rational approximation. *SIAM J. Numer. Anal.*, 20(2):420–436, 1983.

[39] Lloyd N. Trefethen. Circularity of the error curve and sharpness of the CF method in complex Chebyshev approximation. *SIAM J. Numer. Anal.*, 20(6):1258–1263, 1983.

[40] N. J. Young. The singular-value decomposition of an infinite Hankel matrix. *Linear Algebra and its Applications*, 50:639–656, 1983.

Department of Applied Mathematics, University of Colorado, Boulder, CO 80309-0526, United States