2012

TR-2012002: Randomized Matrix Methods for Real and Complex Polynomial Root-Finding

Victor Y. Pan

Guoliang Qian

Ai-Long Zheng

Randomized Matrix Methods for Real and Complex Polynomial Root-finding

Victor Y. Pan[1,2],[a], Guoliang Qian[2],[b], and Ai-Long Zheng[2],[c]
Supported by NSF Grant CCF-1116736 and PSC CUNY Award 64512–0042

[1] Department of Mathematics and Computer Science
Lehman College of the City University of New York
Bronx, NY 10468 USA
[2] Ph.D. Programs in Mathematics and Computer Science
The Graduate Center of the City University of New York
New York, NY 10036 USA
[a] victor.pan@lehman.cuny.edu
http://comet.lehman.cuny.edu/vpan/
[b] gqian@gc.cuny.edu
[c] azheng-1999@yahoo.com

Abstract

To advance the known approach to univariate polynomial root-finding via computations in Frobenius matrix algebra, we incorporate some effective methods for matrix eigen-solving, randomized matrix algorithms, and subdivision techniques. We also develop iterations directed to the approximation of only real roots. Our analysis and experiments show effectiveness of the resulting numerical real and complex root-finders. Our auxiliary results on randomized matrix computations can be of independent interest.

KEYWORDS: Root-finding, Eigen-solving, Rational maps of real matrices, Randomization.

1 Introduction

Polynomial root-finding is the oldest subject of mathematics and computational mathematics, but the list of hundreds if not thousands of the known algorithms grows every year (see [1], [2], [56], [57], [19], [46], [47], [49], [10], [11], [28], [29], [77], [43], [5], [79], [63], [66], [67], [50], [82], [48], [30], [73], [75], [76], and the bibliography therein). Many algorithms are directed to computing a single (e.g., absolutely largest) root (zero) of a polynomial or a subset of all its roots, e.g., all r its real roots. In some applications (e.g., to algebraic geometric optimization) only the r real roots are of interest, and they can be much less numerous than all n roots; nevertheless the best numerical subroutines such as MPSolve approximate all these r real roots about as fast and as slow as all n complex roots.

An important recent direction is root-finding for a polynomial $p(x)$ via eigen-solving for the associated companion matrix C_p; this allows incorporation of the well developed numerical matrix methods in [32], [6], [72], [81], and the bibliography therein. The QR algorithm, adopted for polynomial root-finding by Matlab, avoids numerical problems, faced by many other companion matrix methods [32, Section 7.4.6], but is not readily amenable to exploiting the rich structure of the companion matrix. Extensive research toward such exploitation in QR- and LR-based root-finders has been initiated by [11], [12] and [7] and still goes on (see [5], [79], [4], and the references therein).

The Rayleigh Quotient iteration [32, Section 8.2.2] has no such problems. Its adjustment to polynomial root-finding in [10] and [66] performs every iteration and every deflation step in linear arithmetic time by exploiting the structures of companion and generalized companion matrices.

Our point of departure is the somewhat similar approach of Cardinal [18], [16], [62]. It enhances the Power Method [32, Section 8.2.2] and the method of [68], [69], and [37] by reducing every multiplication in the Frobenius algebra generated by the companion matrix C_p to a small number of FFTs. We obtain further progress by incorporating some advanced techniques of randomization, subdivision and matrix eigen-solving; furthermore we extend this approach to real root-finding.

Presenting our algorithms we compare them with Cardinal's and other relevant methods. For our real numerical root-finding, however, the preceding work seems to be confined to two sections of [66], whose techniques are peripheral to our present constructions.

Polynomial root-finding is fundamental for symbolic computation, but we largely employ fast numerical algorithms with rounding. Consequently, our real root-finders produce both real and nearly real eigenvalues and roots (we specify this class quantitatively), but we can readily select among them the real roots (see Remark 5.3 in Section 5). Our real root-finders are effective as long as both real and nearly real roots together are much less numerous than the other roots.

Overall our analysis and experiments suggest that our approach leads to substantial advance of real and complex polynomial root-finding by means of numerical methods.

Some of our techniques can be of independent interest, e.g., the ones for saving inversions in matrix sign approximation, controlling the norms of the matrices computed in our iterations, computing matrix functions that have dominant eigenspaces associated with real eigenvalues, and the recovery of these eigenspaces by means of randomization techniques.

Our auxiliary estimates for the condition numbers of random $n \times n$ Toeplitz matrices show that these numbers do not tend to grow exponentially in n as $n \to \infty$, in good accordance with our previous and present tests. These estimates are not critical for our present algorithms, but meet a research challenge from [74], contradict a "folk theorem" that states the opposit, and are partly motivated by the study in [9] of some large but special classes of Toeplitz matrices, and have important applications to computations with structured matrices (see [64]).

One can expect to see further advance of our approach, e.g., based on more intricate maps of the complex plane. Another potential resource is the combination with other polynomial root-finders, e.g., the Rayleigh Quotient iteration (cf. Remark 10.1), Newton's iteration (both can be concurrently applied at distinct initial points), and nonnumerical real polynomial root-finders, namely, subdivision and continued fraction methods (see [50], [28], [29], [77], [43], [82], and the bibliography therein). These highly successful algorithms can supply auxiliary information for our computations (e.g., the number of real roots and their bounds) or can be used as complementary techniques handling the inputs that are hard for our numerical treatment.

We organize our presentation as follows. The next section is devoted to definitions and preliminary results. In Sections 3 and 4 we cover randomized matrix computations. In Section 5 we present our Basic Flowcharts. In Section 6 we combine them with repeated squaring to approximate absolutely largest roots as well as the roots closest to a selected complex point. In Section 7 we recall the matrix sign function. In Section 8 we apply it to eigen-solving. We cover its computation, adjust it to real eigen-solving and modify it to save matrix inversions in Sections 9–11. Section 12 covers our numerical tests, which are the contribution of the second and third authors.

2 Definitions and preliminaries

Hereafter "op" stands for "arithmetic operation", "is expected" and "is likely" mean "with a probability near 1", and "small", "large", "close", and "near" are meant in the context. We assume computations in the fields of complex and real numbers $\mathbb{C}$ and $\mathbb{R}$, respectively.

For $\rho' > \rho > 0$ and a complex c, define the circle $\mathcal{C}_\rho(c) = \{\lambda : \ |\lambda - c| = \rho\}$, the disc $\mathcal{D}_\rho(c) = \{\lambda : \ |\lambda - c| \le \rho\}$, and the annulus $\mathcal{A}_{\rho,\rho'}(c) = \{\lambda : \ \rho \le |\lambda - c| \le \rho'\}$.

Matrix computations: fundamentals [32], [71], [80]. $(B_j)_{j=1}^s = (B_1 \mid B_2 \mid \ldots \mid B_s)$ is the $1 \times s$ block matrix with blocks $B_1, B_2, \ldots, B_s$. $\mathrm{diag}(B_j)_{j=1}^s = \mathrm{diag}(B_1, B_2, \ldots, B_s)$ is the $s \times s$ block diagonal matrix with diagonal blocks B_1, B_2,..., B_s.

$I = I_n = (\mathbf{e}_1 \mid \mathbf{e}_2 \mid \ldots \mid \mathbf{e}_n)$ is the $n \times n$ identity matrix with columns $\mathbf{e}_1$, $\mathbf{e}_2, \ldots, \mathbf{e}_n$. $J = J_n = (\mathbf{e}_n \mid \mathbf{e}_{n-1} \mid \ldots \mid \mathbf{e}_1)$ is the $n \times n$ reflection matrix, $J^2 = I$. $O_{k,l}$ is the $k \times l$ matrix filled with zeros.

M^T is the transpose of a matrix M.

$\mathcal{R}(M)$ is the range of a matrix M, that is the linear space generated by its columns. $\mathcal{N}(M) = \{\mathbf{v} : M\mathbf{v} = \mathbf{0}\}$ is its null space, $\mathrm{rank}(M) = \dim(\mathcal{R}(A))$. A matrix of full column rank is a *matrix basis* of its range.

M^+ is the Moore–Penrose pseudo inverse of M [32, Section 5.5.4]. An $n \times m$ matrix $X = M^{(I)}$ is a left (resp. right) inverse of an $m \times n$ matrix M if $XM = I_n$ (resp. if $MY = I_m$). M^+ is an $M^{(I)}$ for a matrix M of full rank; $M^{(I)} = M^{-1}$ for a nonsingular matrix M.

We use the matrix norms $||\cdot||_h$ for $h = 1, 2, \infty$ and write $||\cdot|| = ||\cdot||_2$.

A matrix U is *unitary*, *orthogonal* and *orthonormal* and $||U|| = 1$ if $U^TU = I$.

Theorem 2.1. *[32, Theorem 5.2.2]. A matrix M of full column rank has unique QR factorization $M = QR$ where $Q = Q(M)$ is a unitary matrix and $R = R(M)$ is a square upper triangular matrix with positive diagonal entries.*

Matrix computations: eigenspaces and SVD [32], [72], [80], [81], [6]. $\mathcal{S} \subseteq \mathbb{C}^{n \times n}$ is an *invariant subspace* or *eigenspace* of a matrix $M \in \mathbb{C}^{n \times n}$ if $M\mathcal{S} = \{M\mathbf{v} : \mathbf{v} \in \mathcal{S}\} \subseteq \mathcal{S}$.

Theorem 2.2. *[72, Theorem 4.1.2], [80, Section 6.1], [81, Section 2.1]. Let $U \in \mathbb{C}^{n \times r}$ be a matrix basis for an eigenspace $\mathcal{U}$ of a matrix $M \in \mathbb{C}^{n \times n}$. Then the matrix $L = U^{(I)}MU$ is unique (that is independent of the choice of the left inverse $U^{(I)}$) and satisfies $MU = UL$.*

The above pair $\{L, \mathcal{U}\}$ is an *eigenpair* of a matrix M, L is its *eigenblock* and Φ is the *associated eigenspace* of L [72]. If $L = \lambda I_n$, then $\{\lambda, \mathcal{U}\}$ is also called an eigenpair of a matrix M, in this case $\det(\lambda I - M) = 0$ and $\mathcal{N}(M - \lambda I)$ is the eigenspace associated with the *eigenvalue* λ and made up of its *eigenvectors*. $\Lambda(M)$ is the set of all eigenvalues of M, called its *spectrum*. $\rho(M) = \max_{\lambda \in \Lambda(M)} |\lambda|$ is the *spectral radius* of M. Theorem 2.2 implies that $\Lambda(L) \subseteq \Lambda(M)$. For an eigenpair $\{\lambda, \mathcal{U}\}$ write $\psi = \min |\lambda/\mu|$ over $\lambda \in \Lambda(L)$ and $\mu \in \Lambda(M) - \Lambda(L)$; call the eigenspace $\mathcal{U}$ *dominant* if $\psi > 1$, *dominated* if $\psi < 1$, *strongly dominant* if $1/\psi \approx 0$, and *strongly dominated* if $\psi \approx 0$.

A scalar λ is *nearly real* (within $\epsilon > 0$) if $|\Im(\lambda)| \le \epsilon|\lambda|$.

An $n \times n$ matrix M is called *diagonalizable or nondefective* if SMS^{-1} is a diagonal matrix for some matrix S, e.g., if M has n distinct eigenvalues. A random real or complex perturbation makes the matrix diagonalizable with probability 1. In all our algorithms we assume diagonalizable input matrices.

Theorem 2.3. *(See [38, Theorem 1.13].) $\Lambda(F(M)) = F(\Lambda(M))$ for a square matrix M and a function $F(x)$ defined on its spectrum. Furthermore $(F(\lambda), \mathcal{U})$ is an eigenpair of $F(M)$ if M is diagonalizable and has an eigenpair $(\lambda, \mathcal{U})$.*

$M = S_M \Sigma_M T_M^T$ is an *SVD* of an $m \times n$ matrix M of a rank ρ provided $S_M S_M^T = S_M^T S_M = I_m$, $T_M T_M^T = T_M^T T_M = I_n$, $\Sigma_M = \mathrm{diag}(\widehat{\Sigma}_M, O_{m-\rho,n-\rho})$, $\widehat{\Sigma}_M = \mathrm{diag}(\sigma_j(M))_{j=1}^{\rho}$, $\sigma_j = \sigma_j(M) = \sigma_j(M^T)$ is the jth largest singular value of a matrix M. These values have the minimax property

$$\sigma_j = \max_{\dim(\mathbb{S})=j} \min_{\mathbf{x} \in \mathbb{S},\ ||\mathbf{x}||=1} ||M\mathbf{x}||,\ j = 1, \ldots, \rho, \tag{2.1}$$

where $\mathbb{S}$ denotes linear spaces [32, Theorem 8.6.1]. Note that σ_j^2 is an eigenvalue of M^TM, $\sigma_1 = ||M||$, $\sigma_\rho = 1/||M^+||$, and $\sigma_j = 0$ for $j > \rho$.

Let $\sigma_q > \sigma_{q+1}$. Then $q \le \rho$ and the matrix $T_{q,M} = T(I_q \mid O_{n-q,q})^T$ generates the right leading singular space $\mathbb{T}_{q,M} = \mathcal{R}(T_{q,M})$ associated with the q largest singular values of the matrix M.

$\kappa(M) = \frac{\sigma_1(M)}{\sigma_\rho(M)} = ||M||\ ||M^+|| \ge 1$ is the condition number of a matrix M of a rank ρ. Such a matrix is *ill conditioned* if $\sigma_1(M) \gg \sigma_\rho(M)$; otherwise *well conditioned*. $\kappa(M) = ||M|| = ||M^+|| = 1$ for unitary matrices M.

A matrix M has *numerical rank* ρ if the ratio $\frac{\sigma_1}{\sigma_\rho}$ is not large but if $\frac{\sigma_\rho}{\sigma_{\rho+1}} \ll 1$.

Toeplitz matrices [60, Ch. 2]. An $m \times n$ Toeplitz matrix $T = (t_{i-j})_{i,j=1}^{m,n}$ is defined by the $m+n-1$ entries of its first row and column; e.g.,

$$T = (t_{i-j})_{i,j=1}^{n,n} = \begin{pmatrix} t_0 & t_{-1} & \cdots & t_{1-n} \\ t_1 & t_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & t_{-1} \\ t_{n-1} & \cdots & t_1 & t_0 \end{pmatrix}.$$

We write $T = Z(\mathbf{v})$ if $T\mathbf{e}_1 = \mathbf{v}$ and if $\mathbf{e}_1^T T = t_{11}\mathbf{e}_1^T$, that is if T is a lower triangular Toeplitz matrix defined by its first column $\mathbf{v}$.

$$Z = \begin{pmatrix} 0 & & & & \\ 1 & \ddots & & & \\ & \ddots & \ddots & & \vdots \\ & & \ddots & 0 & \\ & & & 1 & 0 \end{pmatrix}$$

is the $n \times n$ downshift matrix (its entries are zeros except for the first subdiagonal filled with ones). $Z\mathbf{v} = (v_i)_{i=0}^{n-1}$ where $\mathbf{v} = (v_i)_{i=1}^{n}$ and $v_0 = 0$.

We have $Z(\mathbf{v}) = \sum_{i=0}^{n-1} v_{i+1} Z^i$, and hereafter we write $Z(\mathbf{v})^T = (Z(\mathbf{v}))^T$.

Combine the equations $||Z(\mathbf{v})||_1 = ||Z(\mathbf{v})||_\infty = ||\mathbf{v}||_1$ with the bound $||Z(\mathbf{v})||^2 \le ||Z(\mathbf{v})||_1 ||Z(\mathbf{v})|_\infty$ of [32, Corollary 2.3.2] and obtain

$$||Z(\mathbf{v})|| \le ||\mathbf{v}||_1. \tag{2.2}$$

[31] extends Gohberg–Sementsul's celebrated formula as follows.

Theorem 2.4. *Suppose $T_{n+1} = (t_{i-j})_{i,j=0}^{n}$ is a nonsingular Toeplitz matrix, write $(v_i)_{i=0}^{n} = T_{n+1}^{-1}\mathbf{e}_1$, $\mathbf{v} = (v_i)_{i=0}^{n-1}$, $\mathbf{v}' = (v_i)_{i=1}^{n}$, $(w_i)_{i=0}^{n} = T_{n+1}^{-1}\mathbf{e}_{n+1}$, $\mathbf{w} = (w_i)_{i=0}^{n-1}$, and $\mathbf{w}' = (w_i)_{i=1}^{n}$. If $v_n \ne 0$, the matrix $T_{1,0} = (t_{i-j})_{i=1,j=0}^{n,n-1}$ is nonsingular and $v_n T_{1,0}^{-1} = Z(\mathbf{w})Z(J\mathbf{v}')^T - Z(\mathbf{v})Z(J\mathbf{w}')^T$.*

Polynomials and companion matrices. Write

$$p(x) = \sum_{i=0}^{n} p_i x^i = p_n \prod_{j=1}^{n} (x - \lambda_j), \tag{2.3}$$

$$p_{\rm rev}(x) = x^n p(1/x) = \sum_{i=0}^{n} p_i x^{n-i} = p_n \prod_{j=1}^{n} (1 - x\lambda_j), \tag{2.4}$$

$p_{\rm rev}(x)$ is the reverse polynomial of $p(x)$,

$$C_p = \begin{pmatrix} 0 & & & & -p_0/p_n \\ 1 & \ddots & & & -p_1/p_n \\ & \ddots & \ddots & & \vdots \\ & & \ddots & 0 & -p_{n-2}/p_n \\ & & & 1 & -p_{n-1}/p_n \end{pmatrix} = Z - \frac{1}{p_n}\mathbf{e}_n^T \mathbf{p}, \text{ for } \mathbf{p} = (p_j)_{j=0}^{n-1},$$

and $C_{p_{\rm rev}} = JC_pJ$ are the $n \times n$ companion matrices of the polynomials $p(x) = \det(xI_n - C_p)$ and $p_{\rm rev}(x) = \det(xI_n - C_{p_{\rm rev}})$, respectively.

Fact 2.1. *(See [18] or [62].) The companion matrix $C_p \in \mathbb{C}^{n\times n}$ of a polynomial $p(x)$ of (2.3) generates an algebra $\mathcal{A}$ of matrices having structure of Toeplitz type. One needs $O(n)$ ops for addition in $\mathcal{A}$, $O(n \log n)$ ops for multiplication in $\mathcal{A}$, $O(n \log^2 n)$ ops for inversion in $\mathcal{A}$, and $O(n \log n)$ ops for multiplying a matrix from $\mathcal{A}$ by a square Toeplitz matrix.*

3 Ranks and condition numbers of random matrices

3.1 Random variables and random matrices

Definition 3.1. $F_\gamma(y) = Probability\{\gamma \le y\}$ *for a real random variable* γ *is the* cumulative distribution function (cdf) *of* γ *evaluated at* y. $F_{g(\mu,\sigma)}(y) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{y} \exp(-\frac{(x-\mu)^2}{2\sigma^2})dx$ *for a Gaussian random variable* $g(\mu,\sigma)$ *with a mean* μ *and a positive variance* σ^2, *and so*

$$\mu - 4\sigma \le y \le \mu + 4\sigma \text{ with a probability near one.} \tag{3.1}$$

Definition 3.2. $\mathcal{G}_{\mu,\sigma}^{m\times n}$ *is the set of* $m \times n$ Gaussian random matrices *having a mean* μ *and a positive variance* σ^2, *that is matrices filled with independent Gaussian random variables, all sharing these mean and variance. For* $\mu = 0$ *and* $\sigma^2 = 1$ *they are* standard Gaussian random matrices. $\mathcal{T}_{\mu,\sigma}^{m\times n}$ *is the set* $\mathcal{G}_{\mu,\sigma}^{m\times n}$ *restricted to Toeplitz matrices. (With probability 1 matrices* $G \in \mathcal{G}_{\mu,\sigma}^{m\times n}$ *and* $T \in \mathcal{T}_{\mu,\sigma}^{m\times n}$ *have full rank and for* $m = n$ *no entry of the matrices* G^{-1} *and* T^{-1} *vanishes.)*

Definition 3.3. *Suppose* $(\sum_{i=1}^{n} v_i^2)^{1/2} = ||(v_i)_{i=1}^n||$, $(v_i)_{i=1}^n \in \mathcal{G}_{\mu,\sigma}^{n\times 1}$. *Then write* $\chi_{0,1,n}(y) = \frac{2}{2^{n/2}\Gamma(n/2)} \int_{-\infty}^{y} x^{n-1}\exp(-x^2/2)dx$ *for* $y \ge 0$. *Here* $\Gamma(h) = \int_0^\infty x^{h-1}\exp(-x)dx$; $\Gamma(n+1) = n!$ *for integers* $n \ge 0$.

Lemma 3.1. *Suppose* y *is a positive number,* $T \in \mathcal{T}_{\mu,\sigma}^{n\times n}$, j *is an integer,* $1 \le j \le n$, *and* $\mathbf{x}_j \in \mathbb{R}^{n\times 1}$ *is the unit vector orthogonal to all vectors* $T\mathbf{e}_i$ *for* $i \ne j$. *Then*

$$\text{Probability}\{||T^{-1}\mathbf{e}_j|| > 1/y\} \le \text{Probability}\{|\mathbf{x}_j^T T\mathbf{e}_j| < y\}.$$

Proof. Reuse the proof of [74, Lemma 3.2]. □

Lemma 3.2. *[74, Lemma A.2]. For* $\mathbf{t} \in \mathcal{R}^{n\times 1}$, $\mathbf{b} \in \mathcal{G}_{\mu,\sigma}^{n\times 1}$, *and a nonnegative* y, *we have* $F_{|\mathbf{t}^T\mathbf{b}|}(y) = \text{Probability}\{|\mathbf{t}^T\mathbf{b}| \le y\} \le \sqrt{\frac{2}{\pi}}\frac{y}{\sigma}$.

Remark 3.1. *The latter bound, independent of* μ *and* n, *holds even where all coordinates of the vector* $\mathbf{b}$ *are fixed, except for a single coordinate in* $\mathcal{G}_{\mu,\sigma}$.

3.2 Condition numbers of Gaussian random matrices

Gaussian random matrices tend to be well conditioned [22], [26], [27], [20], and actually even the sum $W + M$ for any $W \in \mathbb{R}^{m\times n}$ and $M \in \mathcal{G}_{\mu,\sigma}^{m\times n}$ is expected to be well conditioned unless the ratio $\sigma/||W||$ is large or small [74]. Next we recall some relevant results from [74] for $W = O$.

We first estimate the smallest singular value of a Gaussian random matrix M. Namely the right hand side of the inequality in the following theorem is an upper bound on the probability (the cdf) that this value is at most a scalar y, and this scalar itself can be viewed as a probabilistic lower bound on the smallest singular value of M, equal to $1/||M^+||$. The proof employs the following lemma.

Lemma 3.3. *(See [74, the proof of Lemma 3.2].) Suppose* y *is a positive number,* $\mathbf{w} \in \mathbb{R}^{n\times 1}$ *is any fixed real unit vector,* $||\mathbf{w}|| = 1$, $M \in \mathcal{G}_{\mu,\sigma}^{n\times n}$, Q *is a unitary matrix such that* $Q\mathbf{w} = \mathbf{e}_1$, $B = QM = (\mathbf{b}_1 \mid \ldots \mid \mathbf{b}_n)$, $\mathbf{t}^T\mathbf{b}_i = 0$ *for* $i = 2, \ldots, n$, *and* $||\mathbf{t}|| = 1$. *Then*

$$\text{Probability}\{||M^{-1}\mathbf{w}|| > y\} \le \max_{\mathbf{b}_1,\ldots,\mathbf{b}_n} \text{Probability}\{|\mathbf{t}^T\mathbf{b}_1| < 1/y\}.$$

Theorem 3.1. *Let* $l = \min\{m, n\}$, $y \ge 0$, $M \in \mathcal{G}_{\mu,\sigma}^{m\times n}$. *Then* M *has full rank with probability 1 and* $F_{1/||M^+||}(y) \le 2.35\, y\sqrt{l}/\sigma$.

Proof. Deduce from the minimax property (2.1) that

$$\sigma_j(M) \ge \sigma_j(C) \text{ for all } j \tag{3.2}$$

if C is a submatrix of a matrix M. Therefore it is sufficient to prove the claimed bound on $F_M(y)$ in the case where $m = n$, and in this case the theorem turns into [74, Theorem 3.3], proved based on Lemmas 3.2 and 3.3. □

The following two theorems supply lower bounds on the probabilities that $||M|| \le y$ and $\kappa(M) = ||M||\ ||M^+|| \le y$ for a scalar y and a Gaussian random matrix M. The arguments y of the cdfs can be viewed as probabilistic upper bounds on the norm $||M||$ and the condition number $\kappa(M)$, respectively.

Theorem 3.2. *(See [25, Theorem II.7]). Suppose $M \in \mathcal{G}_{0,\sigma}^{m\times n}$, $l = \min\{m,n\}$ and $y \ge 2\sigma\sqrt{l}$. Then $F_{||M||}(y) \ge 1 - \exp(-(y-2\sigma\sqrt{l})^2/(2\sigma^2))$.*

The following theorem implies that the function $1 - F_{\kappa(M)}(y)$ decays as $y \to \infty$, and the decay is inversely proportional to $y/\sqrt{\log y}$.

Theorem 3.3. *(See [74, Theorem 3.1]). Suppose $0 < \sigma \le l = \min\{m,n\}$, $\sigma \le 1$, $y \ge 1$, $M \in \mathcal{G}_{0,\sigma}^{m\times n}$ and therefore has full rank with probability one. Then $F_{\kappa(M)}(y) \ge 1 - (14.1 + 4.7\sqrt{(2\ln y)/n})n/(\sigma y)$.*

y is a probabilistic upper bound on $\kappa(M)$. The lower bound on the cdf of $\kappa(M)$ increases as the value σ increases. For small values σy and a fixed n the lower bound becomes negative, in which case the theorem becomes trivial.

Theorem 3.3 is proved in [74] based on combining Theorems 3.2 and 3.1.

3.3 Condition numbers of random Toeplitz matrices

Next we estimate the condition number $\kappa(T_n) = ||T_n||\ ||T^{-1}||$ for $T_n = (t_{i-j})_{i,j=1}^n \in \mathcal{T}_{\mu,\sigma}^{n\times n}$ (cf. empirical data in [64, Table 1], [65]). We have $T = Z(\mathbf{u}) + Z(\mathbf{v})^T$ for $\mathbf{u} = T\mathbf{e}_1$ and $\mathbf{v} = T^T\mathbf{e}_1 - t_0\mathbf{e}_1$. Combine this equation and bound (2.2) to obtain $F_{||T_n||}(y) \ge \chi_{\mu,\sigma,n}(y/2)$.

It remains to bound the norm $||T_n^{-1}||$.

Theorem 3.4. *Under the assumptions of Theorem 2.4, let $T_{n+1} \in \mathcal{T}_{\mu,\sigma}^{(n+1)\times(n+1)}$ and $y \ge 0$. Then $||v_n T_{1,0}^{-1}|| \le 2\alpha\beta$ for $F_{\min\{\alpha,\beta\}}(1/y) \le \sqrt{\frac{2n+2}{\pi}}\frac{y}{\sigma}$.*

Proof. Note that each of the vectors $T_{n+1}\mathbf{e}_1$ and $T_{n+1}\mathbf{e}_{n+1}$ has an entry not shared with the other entries of T_{n+1}, recall Remark 3.1, and deduce that

$$\text{Probability}\{|\mathbf{x}_j^T T_{n+1}\mathbf{e}_j| < y\} \le \sqrt{\frac{2}{\pi}}\frac{y}{\sigma} \text{ for } \mathbf{x}_j \text{ of Lemma 3.1, } j = 1 \text{ or } j = n+1.$$

Combine this bound, Theorem 2.4, Lemma 3.1, and the inequalities (2.2) and $||\mathbf{v}|| \le \sqrt{n}||\mathbf{v}||_1$. □

Note that $\frac{1}{|v_n|} = |\frac{\det T_{n+1}}{\det T_{0,1}}| = |\frac{\det T_{n+1}}{\det T_n}|$ for $T_k \in \mathcal{T}_{\mu,\sigma}^{k\times k}$ and that Hadamard's inequality bounds the geometric mean $(\prod_{k=1}^n |\frac{\det T_{k+1}}{\det T_k}|)^{\frac{1}{n}} = \frac{1}{t}|\det T_{n+1}|^{\frac{1}{n}} \le (n+1)^{\frac{1}{2}(1+\frac{1}{n})}t$ provided $t \ge \max_{i,j,k} |\mathbf{e}_i^T T_k \mathbf{e}_j|$. In our case it is extremely unlikely that t exceeds $|\mu| + 4n\sigma$ for $k \le n+1$.

4 Condition numbers of randomized matrix products

We wish to bound the condition number $\kappa(MG) = ||MG||\ ||(MG)^+||$ of the matrix products of fixed matrix M and Gaussian random matrix G. Since $||MG|| \le ||M||\ ||G||$, we just need to extend the estimates of Theorem 3.1 to probabilistic lower bounds on the smallest singular values of the products of fixed and random matrices.

Theorem 4.1. *Suppose $M \in \mathcal{G}_{\mu,\sigma}^{m\times n}$, $r(M) = \text{rank}(M) \ge r$, $G \in \mathcal{G}^{r\times m}$. Then the matrix M has full rank r with probability 1 and $F_{1/||(MG)^+||}(y) \le 2.35y\sqrt{r(M)}/(\sigma_{r(M)}(M)\sigma)$.*

The theorem implies that $\sigma_{\text{rank}(MG)} = 1/||(MG)^+|| \le y$ with a probablilty of at most the order y, and so it is unlikely that multiplication by a square or rectangular Gaussian random matrix can dramatically decrease the smallest positive singular value of a matrix, although $UV = O$ for some pairs of rectangular unitary matrices U and V.

Our proof of Theorem 4.1 employs the following three lemmas. The first two of them are immediately implied by minimax property (2.1).

Lemma 4.1. *Suppose* $M = \mathrm{diag}(\sigma_i)_{i=1}^n$, $G \in \mathbb{R}^{n\times r}$, $\mathrm{rank}(M) = n$, $\mathrm{rank}(G) = r(G)$. *Then* $\mathrm{rank}(MG) = r(G)$ *and* $\sigma_j(MG \geq \sigma_j(G)\sigma_n(M)$ *for all* j.

Lemma 4.2. $\sigma_j(GM) = \sigma_j(MH) = \sigma_j(M)$ *for all* j *if* G *and* H *are square unitary matrices.*

Lemma 4.3. *[74, Proposition 2.2]. Suppose* $W \in \mathcal{G}_{\mu,\sigma}^{m\times n}$, $SS^T = S^TS = I_m$, $TT^T = T^TT = I_n$. *Then* $SW \in \mathcal{G}_{\mu,\sigma}^{m\times n}$ *and* $WT \in \mathcal{G}_{\mu,\sigma}^{m\times n}$.

Proof of Theorem 4.1. Let $M = S_M\Sigma_M T_M^T$ be SVD where $\Sigma_M = \mathrm{diag}(\widehat{\Sigma}_M, O) = \widehat{\Sigma}_M\,\mathrm{diag}(I_{r(M)}, O)$, $\widehat{\Sigma}_M = \mathrm{diag}(\sigma_j(M))_{j=1}^{r(M)}$. Write $G_{r(M)} = \mathrm{diag}(I_{r(M)}, O)T_M^TG$, and so $\Sigma_M T_M^TG = \widehat{\Sigma}_M G_{r(M)}$.

We have $MG = S_M\Sigma_M T_M^TG$, and so $\sigma_j(MG) = \sigma_j(\Sigma_M T_M^TG)$ for all j by virtue of Lemma 4.2 (since S_M is a square unitary matrix). Substitute $\Sigma_M T_M^TG = \widehat{\Sigma}_M G_{r(M)}$ and obtain that $\sigma_j(MG) = \sigma_j(\widehat{\Sigma}_M G_{r(M)})$. Now, by virtue of Lemma 4.1 we have $\sigma_j(MG) = \sigma_j(\widehat{\Sigma}_M G_{r(M)}) \geq \sigma_{r(M)}(M)\sigma_j(G_{r(M)})$ for all j. For $j = r(M)$ obtain

$$\sigma_{r(M)}(MG) \geq \sigma_{r(M)}(M)\sigma_{r(M)}(G_{r(M)}). \tag{4.1}$$

We have $T_M^TG \in \mathcal{G}_{\mu,\sigma}^{m\times n}$ by virtue of Lemma 4.3, since T_M is a square unitary matrix; consequently $G_{r(M)} \in \mathcal{G}_{\mu,\sigma}^{r(M)\times n}$. To complete the proof, estimate $F_{\sigma_{r(M)}(G_{r(M)})}(y) = F_{1/||G_{r(M)}^+||}(y)$ by applying Theorem 3.1 for M replaced by $G_{r(M)}$ and combine this estimate with bound (4.1).

Corollary 4.1. *Define* m, n, G *and* M *as in Theorem 4.1, write* $l = \min\{m, n\}$, *and choose two scalars* y *and* z *such that* $y > 0$ *and* $z \geq 2\sigma\sqrt{l}$. *Then we have*
$F_{\kappa(MG)}(||M||yz) \geq 2 - \exp(-\frac{(z-2\sigma\sqrt{l})^2}{2\sigma^2}) - 2.35y\frac{\sqrt{r(M)}}{\sigma_{r(M)}(M)\sigma}$.

Proof. Combine Theorems 3.2 for $y = z$ and 4.1. □

By setting to zero all singular values of G except for its j largest ones, we reduce its rank to j for any $j < r(M)$. This implies the following extension of the theorem.

Corollary 4.2. *Under the assumptions of Theorem 4.1 we have that*
$F_{\sigma_j(MG)}(y + ||G||\sigma_{j+1}(M)) \leq 2.35y\sqrt{j})/(\sigma_j(M)\sigma)$ *for* $j = 1, 2, \ldots, r(M)$.

Remark 4.1. *The corollary implies a probabilistic bound on the residual norm of the approximation by* $\mathcal{R}(MG)$ *of the leading singular space* $\mathbb{T}_{q,M}$ *of a real* $r \times m$ *matrix* M *having numerical rank* q *where* $G \in \mathcal{G}_{\mu,\sigma}^{m\times q}$ *and, say* $\mu = 0$ *and* $\sigma = 1$. *In the case of a small positive* q *we obtain a low-rank approximation of the matrix* M *(cf. [35], [34], [33] and [39]). In this and other applications of the corollary we can probabilistically bound the norm* $||G||$ *based on Theorem 3.2; furthermore we can repeat generation of Gaussian random matrices* G *until we arrive at a matrix having a sufficiently small norm.*

Remark 4.2. *Since* $(G^TM^T)^T = MG$ *and* $\sigma_j(M) = \sigma_j(M^T)$ *for all* j, G *and* M, *we can immediately extend Theorem 4.1 to pre-multiplication by Gaussian random matrices and respectively extend its corollaties and the latter remark.*

5 Approximating selected eigenvalues and Basic Flowcharts

Next we apply Theorems 2.2 and 2.3 to approximate a specified set $\widehat{\Lambda}$ of the eigenvalues of a matrix (e.g., the set of its absolutely largest or real eigenvalues).

Flowchart 5.1. Reduction of the input size in eigen-solving for a subset of the spectrum.

INPUT: *a diagonalizable matrix* $M \in \mathbb{R}^{n\times n}$ *and a property specifying a subset* $\widehat{\Lambda}$ *of its unknown spectrum.*

OUTPUT: *a pair of matrices* $\{\widehat{L}, \widehat{U}\}$ *that closely approximates an eigenpair* $\{L, \mathcal{U}\}$ *of* M *such that* $\Lambda(L) = \widehat{\Lambda}$.

COMPUTATIONS:

1. *Compute a matrix function* $F(M)$ *that has strongly dominant eigenspace* $\mathcal{U}$*, shared with* M.
2. *Compute and output a matrix* $\widehat{U}$ *of full column rank whose range approximates the eigenspace* $\mathcal{U}$.
3. *Compute the left inverse* $\widehat{U}^{(I)}$.
4. *Compute and output the matrix* $\widehat{L} = \widehat{U}^{(I)} M \widehat{U}$.

At Stage 2 of the flowchart one can apply rank revealing QR or LU factorization of the matrix $F(M)$ [40], [58] (see some other relevant techniques in [35], [34], [33], [39], [65]).

Given an upper bound r_+ on the dimension r of the eigenspace $\mathcal{U}$, we can alternatively employ a randomized multiplier as follows (cf. [65]).

Flowchart 5.2. Randomized approximation of a dominant eigenspace.

INPUT: *a positive integer* r_+ *and a diagonalizable matrix* $F \in \mathbb{R}^{n \times n}$ *that has numerical rank* $n - r$ *and has strongly dominant eigenspace* $\mathcal{U}$ *of dimension* $r > 0$ *for an unknown* $r \le r_+$.

OUTPUT: *an* $n \times r$ *matrix* $\widehat{U}$ *such that* $\mathcal{R}(\widehat{U}) \approx \mathcal{U}$.

COMPUTATIONS:

1. *Compute the* $n \times r_+$ *matrix* FG *for* $G \in \mathcal{G}_{0,1}^{n \times r_+}$.
2. *Compute its rank revealing QR or LU factorization, which outputs its orthogonal matrix basis* $\widehat{U}$.

Let us prove correctness of the flowchart assuming that the matrix F is normal or nearly normal (cf. [32, Section 7.1.3]). Clearly, $\operatorname{rank}(FG) = n - r$ with probability 1. Define the matrix $\tilde{F}$ by zeroing the r smallest singular values of F. We have $\tilde{F} \approx F$ because $\sigma_{n-r+1}(F)$ is small; therefore $\mathcal{R}(\tilde{F}) \approx \mathcal{U}$ and $\tilde{F}G \approx FG$. Deduce from Theorem 4.1 that $\mathcal{R}(\tilde{F}G) \approx \mathcal{R}(\tilde{F})$. Finally combine all these relationships and obtain that $\mathcal{R}(\tilde{F}G) \approx \mathcal{U}$. Normality of F implies the transitivity of the relationship $\approx$.

Remark 5.1. *If* $F = F(C_p)$ *and the integer* r_+ *is not small, we can choose matrix* $G \in \mathcal{T}^{n \times r_+}$ *and avoid computing QR or LU factorization at Stage 2; then we would multiply* F *by* G *in* $O(n \log n)$ *ops (see Fact 2.1). Theorem 3.4 is necessary but not sufficient for proving extension of Theorem 4.1 to the case of Toeplitz matrix* G*; Table 12.1, however, supports such an extension empirically.*

In some cases we naturally arrive at matrices $\tilde{F}(M)$ having dominated (rather than dominant) eigenspaces $\mathcal{U}$. If the matrix $\tilde{F}(M)$ is nonsingular, then $\mathcal{U}$ is a dominant eigenspace of the matrix $(\tilde{F}(M))^{-1}$, and we can apply Stages 2–4 of Flowchart 5.1 to this eigenspace. Alternatively, we can employ the following variation of Flowchart 5.1.

Flowchart 5.3. Dual reduction of input size in eigen-solving for a subset of the spectrum.

INPUT, OUTPUT *and Stages 3 and 4 of* COMPUTATIONS *as in Flowchart 5.1.*

COMPUTATIONS:

1. *Compute a matrix function* $\tilde{F}(M)$ *that has strongly dominated eigenspace approximating* $\mathcal{U}$.

2. *Apply the Inverse Orthogonal Iteration [32, page 339] to the matrix $\tilde{F}(M)$ to output a matrix $\widehat{U}$ of full column rank whose range approximates the eigenspace $\mathcal{U}$. Output $\widehat{L} = \widehat{U}^{(I)} M \widehat{U}$.*

Remark 5.2. *Seeking a single eigenvalue of M and having performed Stage 1 of Flowchart 5.1 (resp. 5.3), we can apply the Power (resp. Inverse Power) Method (cf. [32, Sections 7.3.1 and 7.6.1], [10]) to approximate an eigenvector $\mathbf{v}$ of the matrix $F(M)$ in its dominant (resp. dominated) eigenspace $\mathcal{U}$. This eigenvector is shared with M by virtue of Theorem 2.3, and we can approximate the associated eigenvalue of M by the Rayleigh quotient $\mathbf{v}^T M\mathbf{v}/\mathbf{v}^T\mathbf{v}$ or a simpler quotient in [10] and [66]. We can employ deflation or use other initial approximations (cf. our Section 9.3 and [42]) to approximate other eigenvalues of M.*

Remark 5.3. *In numerical implementation of the flowchart we compute a dominant (resp. dominated) eigenspace $\mathcal{U}_+$ of the matrix $F_+(M)$ (resp. $\tilde{F}_+(M)$) such that $\mathcal{U}_+ \supseteq \mathcal{U}$ and has a dimension $r_+ \geq r$. The output matrix L_+ has size $r_+ \times r_+$ and can share with M some extraneous eigenvalues. E.g., in numerical real eigen-solving the eigenspace $\mathcal{U}_+$ is associated with all real and nearly real eigenvalues of M; having computed all eigenvalues of L_+, we can readily select from them the real ones.*

In the next sections we describe some algorithms for computing matrix functions $F(M)$ and $\tilde{F}(M)$ at Stages 1 of Flowcharts 5.1 and 5.3.

6 Repeated squaring

Theorem 2.3 for $F(M) = M^k$ implies that for a diagonalizable matrix M and sufficiently large integers k, the matrices M^k have dominant eigenspace $\mathcal{U}$ associated with the set of the absolutely largest eigenvalues of M. For a fixed or random real or complex shift s we can write $M_0 = M - sI$ and compute $M_0^{2^h}$ in h squarings,

$$M_{h+1} = a_h M_h^2,\ a_h \approx 1/||M_h||^2 \text{ for } h = 0, 1, \ldots \tag{6.1}$$

Suppose M is a real diagonalizable matrix with simple eigenvalues; then with probability 1 the dominant eigenspace $\mathcal{U}$ of M_{h^2} has dimension 1 for random nonreal shifts s and dimension 1 or 2 for random real s.

For $M = C_p$ we can follow [18] and apply the FFT-based algorithms that support Fact 2.1 to perform every squaring and every multiplication in $O(n \log n)$ ops. The bottleneck of an algorithm in [18] for $M = C_p$ is the recovery of the roots of $p(x)$ at the end of the squaring process where $|\lambda_j| \approx |\lambda_k|$ for $j \neq k$. [62] relieves some difficulties based on approximating the roots of $p'(x)$, $p''(x)$, etc., but the techniques of [62] are still too close to the symbolic recovery methods of [18]. In contrast Flowcharts 5.1 and 5.3 reduce the computations of the r eigenvalues of a selected subset of the spectrum $\Lambda(M)$ to eigen-solving for the $r \times r$ matrix L, which is a simple task for small r.

Now replace M_0 in (6.1) by $M_0 = (M - \sigma I)^{-1}$ for a fixed complex σ. Then the dominant eigenspace of M_h for large h is associated with the set of the eigenvalues of M that are the nearest to σ, e.g., the absolutely smallest eigenvalues where $\sigma = 0$. For $M = C_p$ we can alternatively write $M_0 = C_{p_{\text{rev}}(x-\sigma)}$ in (6.1).

7 Matrix sign function and dominant eigenspaces

Definition 7.1. *For two real numbers $x \neq 0$ and y, the function* $\text{sign}(x + y\sqrt{-1})$ *is equal to* 1 *if $x > 0$ and is equal to* -1 *if $x < 0$.*

Definition 7.2. *(See [38].) Let $A = ZJZ^{-1}$ be a Jordan canonical decomposition of an $n \times n$ matrix A where $J = \text{diag}(J_-, J_+)$, J_- is a $p \times p$ matrix and all its p diagonal entries have negative real parts, whereas J_+ is a $q \times q$ matrix and all its q diagonal entries have positive real parts. Then* $\text{sign}(A) = Z\,\text{diag}(-I_p, I_q)Z^{-1}$. *Equivalently* $\text{sign}(A) = A(A^2)^{-1/2}$ *or* $\text{sign}(A) = \frac{2}{\pi} A \int_0^\infty (t^2 I_n + A^2)^{-1} dt$.

Definition 7.3. *Assume the matrices $A = ZJZ^{-1}$, J_- and J_+ above, except that $n = p+q+r$ and $J = \mathrm{diag}(J_-, J_0, J_+)$ for a $r \times r$ matrix J_0 whose all r diagonal entries have real parts 0. Then fix some $r \times r$ real diagonal matrix D_r, e.g., $D_r = O_{r,r}$, and define a* generalized matrix sign function $\mathrm{sign}(A)$ *by writing* $\mathrm{sign}(A) = Z\,\mathrm{diag}(-I_p, D_r\sqrt{-1}, I_q)Z^{-1}$.

We have the following simple results.

Theorem 7.1. *Assume the generalized matrix sign function* $\mathrm{sign}(A)$ *defined for an $n \times n$ matrix $A = ZJZ^{-1}$. Then for some real $r \times r$ diagonal matrix D_r we have*

$$I_n - \mathrm{sign}(A) = Z^{-1}\,\mathrm{diag}(2I_p, I_r - D_r\sqrt{-1}, O_{q,q})Z,$$
$$I_n + \mathrm{sign}(A) = Z^{-1}\,\mathrm{diag}(O_{p,p}, I_r + D_r\sqrt{-1}, 2I_q)Z,$$
$$I_n - \mathrm{sign}(A)^2 = Z^{-1}\,\mathrm{diag}(O_{p,p}, I_r + D_r^2, O_{q,q})Z.$$

Corollary 7.1. *Under the assumptions of Theorem 7.1 the matrix $I_n - \mathrm{sign}(A)^2$ has dominant eigenspace of dimension r associated with the eigenvalues of the matrix A that lie on the imaginary axis $\mathcal{IA} = \{\lambda : \Re(\lambda) = 0\}$, whereas the matrices $I_n - \mathrm{sign}(A)$ (resp. $I_n + \mathrm{sign}(A)$) have dominant eigenspaces associated with the eigenvalues of A that either lie on the left (resp. right) of the axis $\mathcal{IA}$ or lie on this axis and have nonzero images in $I_n - \mathrm{sign}(A)$ (resp. $I_n + \mathrm{sign}(A)$).*

8 Eigen-solving via matrix sign computation

Having the matrices A and $F(A) = I_n - \mathrm{sign}(A)^2$ available, we can apply Flowchart 5.1 to approximate the eigenvalues of A that lie on the axis $\mathcal{IA}$. In the next sections we devise real eigen-solvers for a real $n \times n$ matrix M, based on applying these techniques to the matrix $A = M\sqrt{-1}$. Likewise, having the matrices A and $F(A) = I_n - \mathrm{sign}(A)$ (resp. $F(A) = I_n + \mathrm{sign}(A)$) available, we can apply Flowchart 5.1 to approximate all eigenvalues of A that lie either on the axis $\mathcal{IA}$ or on the left (resp. right) from it.

The computed square matrices L have dimensions p_+ and q_+, respectively, where $p \le p_+ \le p+r$ and $q \le q_+ \le q+r$. If $M = C_p$ and if the integer p_+ or q_+ is large, we split out a high degree factor of the polynomial $p(x)$. This can lead to dramatic growth of the coefficients, e.g., in the case where we split the polynomial $x^n + 1$ into the product of two high degree factors, one of them having only roots with positive real parts. The subdivision techniques (cf. [59]) based on the following simple fact, however, give us a universal remedy, unlike the limited remedies in [18].

Fact 8.1. *Suppose $\mathcal{U}$ and $\mathcal{V}$ are two eigenspaces of A and $\Lambda(\mathcal{U})$ and $\Lambda(\mathcal{V})$ are the sets of the associated eigenvalues. Then $\Lambda(\mathcal{U}) \cap \Lambda(\mathcal{V})$ is the set of the eigenvalues of A associated with the eigenspace $\mathcal{U} \cap \mathcal{V}$.*

By computing the matrix sign function of the matrices $\alpha A - \sigma I$ for various selected pairs of complex scalars α and σ, we can define the eigenspace of A associated with the eigenvalues lying in a selected region on the complex plane bounded by straight lines, e.g., in any fixed rectangle with four pairs $\{\alpha, \sigma\}$ where α equals 1 and $\sqrt{-1}$ and $\sigma = k2^l$ for proper integers k and l. By including matrix inversions into this game, we define the eigenvalue regions bounded by straight lines, their segments, circles and their arcs.

9 Iterative algorithms for the matrix sign computation

9.1 Some known algorithms and their convergence

[38, equations (6.17)–(6.20)] define effective iterative algorithms for the square root function $B^{1/2}$; one can readily extend them to $\mathrm{sign}(A) = A(A^2)^{-1/2}$. [38, Chapter 5] presents a number of effective algorithms devised directly for the matrix sign function. Among them we recall Newton's iteration

$$N_0 = A,\ N_{i+1} = (N_i + N_i^{-1})/2,\ i = 0, 1, \ldots, \tag{9.1}$$

based on the Möbius transform $x \to (x + 1/x)/2$, and the [2/0] Padé iteration

$$N_0 = A,\ N_{i+1} = (15I_n - 10N_i^2 + 3N_i^4)N_i/8,\ i = 0, 1, \dots \tag{9.2}$$

Theorem 2.3 implies the following simple corollary.

Corollary 9.1. *Assume iterations (9.1) and (9.2) where neither of the matrices N_i is singular. Let $\lambda = \lambda^{(0)}$ denote an eigenvalue of the matrix N_0 and define*

$$\lambda^{(i+1)} = (\lambda^{(i)} + (\lambda^{(i)})^{-1})/2,\ i = 0, 1, \dots, \tag{9.3}$$

$$\lambda^{(i+1)} = \lambda^{(i)}(15 - 10(\lambda^{(i)})^2 + 3(\lambda^{(i)})^4)/8,\ i = 0, 1, \dots \tag{9.4}$$

Then $\lambda^{(i)} \in \Lambda(N_i)$ for $i = 1, 2, \dots$ provided the pairs $\{N_i, \lambda^{(i)}\}$ are defined by the pairs of equations (9.1), (9.3) or (9.2), (9.4), respectively.

Corollary 9.2. *In iterations (9.3) and (9.4) the images $\lambda^{(i)}$ of an eigenvalue λ of the matrix N_0 for all i lie on the imaginary axis $\mathcal{IA}$ if so does λ.*

By virtue of the following theorems, the sequences $\{\lambda^{(0)}, \lambda^{(1)}, \dots\}$ defined by equations (9.3) and (9.4) converge to ± 1 exponentially fast right from the start. The convergence is quadratic for sequence (9.3) where $\Re(\lambda) \neq 0$ and cubic for sequence (9.4) where $|\lambda - \text{sign}(\lambda)| \leq 1/2$.

Theorem 9.1. *(See [38], [16, page 500].) Write $\lambda = \lambda^{(0)}$, $\delta = \text{sign}(\lambda)$ and $\gamma = |\frac{\lambda-\delta}{\lambda+\delta}|$. Assume (9.3) and $\Re(\lambda) \neq 0$. Then $|\lambda^{(i)} - \delta| \leq \frac{2\gamma^{2^i}}{\gamma^{2^i}+\delta}$ for $i = 0, 1, \dots$.*

Theorem 9.2. *Write $\delta_i = \text{sign}(\lambda^{(i)})$ and $\gamma_i = |\lambda^{(i)} - \delta_i|$ for $i = 0, 1, \dots$. Assume (9.4) and $\gamma_0 \leq 1/2$. Then $\gamma_i \leq \frac{32}{113}(\frac{113}{128})^{3^i}$ for $i = 1, 2, \dots$*

Proof. We clarify the proof of [16, Proposition 4.1]. First verify that $\gamma_{i+1} = \gamma_i^3|3(\lambda^{(i)})^2 + 9\lambda^{(i)} + 8|/8$ and therefore $\gamma_{i+1} \leq \frac{113}{32}\gamma_i^3$ for $i = 0, 1, \dots$. Now the claimed bounds follow by induction on i because $\gamma_0 \leq 1/2$. □

9.2 Variants for real eigen-solving

As we mentioned we can reduce real eigen-solving for a real matrix M to matrix sign computation for $A = M\sqrt{-1}$, but next we substitute $N_0 = M$ in lieu of $N_0 = A$ into matrix sign iterations (9.1) and (9.2) and equilvalently rewrite them to avoid involving nonreal values,

$$N_0 = M,\ N_{i+1} = 0.5(N_i - N_i^{-1}) \text{ for } i = 0, 1, \dots, \tag{9.5}$$

$$N_0 = M,\ N_{i+1} = -(3N_i^5 + 10N_i^3 + 15N_i)/8 \text{ for } i = 0, 1, \dots. \tag{9.6}$$

Now the matrices N_i and the images $\lambda^{(i)}$ of every real eigenvalue λ of M are real for all i, whereas the results of Theorems 9.1 and 9.2 are immediately extended. The images of every nonreal λ converge to $\text{sign}(\Im(\lambda))\sqrt{-1}$ quadratically under (9.5) if $\Re(\lambda) \neq 0$ and cubically under (9.6) if $\lambda \in \mathcal{D}_{1/2}(\text{sign}(\Im(\lambda))\sqrt{-1})$.

Under the maps $M \to I_n + N_i^2$ for N_i in the above iterations, the images $1 + (\lambda^{(i)})^2$ of nonreal eigenvalues λ of M in the respective basins of convergence converge to 0, whereas for real λ the images are real and are at least 1 for all i. Thus for sufficiently large integers i we yield strong domination of the eigenspace of N_i associated with the images of real eigenvalues of M.

9.3 Newton's iteration with shifts for real matrix sign function

Iteration (9.5) fails where for some i the matrix N_i is singular or nearly singular, that is has eigenvalue 0 or near 0, but then we can approximate this eigenvalue by applying the Rayleigh Quotient Iteration [32, Section 8.2.3], [10] or the Inverse Orthogonal Iteration [32, page 339].

If we seek other real eigenvalues as well, we can deflate the matrix M and apply Flowchart 5.1 to the resulting matrix of a smaller size. Alternatively we can apply it to the matrix $N_i + \rho_i I_n$ for a shift ρ_i randomly generated in the range $-r \le \rho_i \le r$ for a positive r. We choose r reasonably small and can expect both avoiding degeneracy and, by virtue of Theorems 9.2 7.1, having the images of all nonreal eigenvalues of M still rapidly converging to a small neighborhood of the points $\pm\sqrt{-1}$, thus ensuring their isolation from the images of real eigenvalues.

9.4 Controlling the norms in the [2/0] Padé iterations

We have no singularity problem with iteration (9.6), but have numerical problems where the norms $||N_i||$ grow large. If the nonreal eigenvalues of the matrix N_0 lie in the two discs $\mathcal{D}_{1/2}(\pm\sqrt{-1})$, then their images also stay there by virtue of extension of Theorem 9.2, and then the norms $||N_i||$ can be large only where some real eigenvalues of the matrices N_i are absolutely large.

Now suppose the nonreal eigenvalues of M have been mapped into the two discs $\mathcal{D}_{y_i}(\pm\sqrt{-1})$ for $0 < y_i < 0.1$. (One or two steps (9.6) move every $\mu \in \mathcal{D}_{1/2}(\pm\sqrt{-1})$ into the discs $\mathcal{D}_{y_i}(\pm\sqrt{-1})$, cf. Theorem 9.2.) Then the transformation $N_i \to N_i(N_i^2 + 2I_n)^{-1}$ confronts excessive norm growth by mapping all real eigenvalues of N_i into the range $[-\frac{1}{4}\sqrt{2}, \frac{1}{4}\sqrt{2}]$ and mapping all nonreal eigenvalues of N_i into the discs $\mathcal{D}_{w_i}(\pm\sqrt{-1})$ for $w_i \le \frac{1+y_i}{1-2y_i-y_i^2}$. E.g., $w_i < 0.4$ for $y_i = 0.1$, whereas $w_i < 0.17$ for $y_i = 0.05$, and then single step (9.6) would more than compensate for such a minor dilation of the discs $\mathcal{D}_{y_i}(\pm\sqrt{-1})$ (see Theorem 9.2).

10 Modifications with fewer matrix inversions

We should apply iteration (9.6) rather than (9.5) to exploit its cubic convergence and to avoid matrix inversions as soon as the images of the targited eigenvalues λ of M have been moved into the discs $\mathcal{D}_{1/2}(\pm\sqrt{-1})$. Our goal is to achieve this in fewer steps (9.5) based on nonreal computations and repeated squaring (6.1) for appropriate matrices M_0.

10.1 Mapping the real line onto unit circle and repeated squaring

Next we incorporate repeated squaring of a matrix between its back and forth transforms defined by the maps of the complex plane $\mu \to \lambda$ and $\lambda \to \mu$ below.

Fact 10.1. *Write* $\lambda = u + v\sqrt{-1}$,

$$\mu = (a\lambda + \sqrt{-1})(a\lambda - \sqrt{-1})^{-1},\ \beta_k = \frac{\sqrt{-1}(\mu^k + 1)}{a(\mu^k - 1)} \tag{10.1}$$

for a positive integer k *and a real* $a \neq 0$ *(one can simply choose* $a = 1$*, but other choices can be more effective). Then*

(a) $\lambda = \frac{\sqrt{-1}(\mu+1)}{a(\mu-1)}$,

(b) $\mu = \frac{n(\lambda)}{d(\lambda)}$ *for* $n(\lambda) = u^2 + v^2 - a^2 = 2au\sqrt{-1}$ *and* $d(\lambda) = u^2 + (v-a)^2)^2$*, and consequently*

(c) $|\mu|^2 = \frac{u^2+(v+a)^2}{u^2+(v-a)^2} = 1 + \frac{4av}{u^2+(v-a)^2}$,

(d) $|\mu| = 1$ if and only if λ is real.

Furthermore

(e) $\beta_k = \frac{n_k(\lambda)}{d_k(\lambda)}$ *for* $n_k(\lambda) = \sum_{g=0}^{\lfloor k/2 \rfloor}(-1)^g \binom{k}{2g} (a\lambda)^{k-2g}$ *and*

$$d_k(\lambda) = a \sum_{g=0}^{\lfloor k/2 \rfloor} (-1)^{g+1} \binom{k}{2g+1} (a\lambda)^{h-2g-1}.$$

Fact 10.1 implies that the transform $\lambda \to \mu$ maps the real line onto the unit circle $\mathcal{C}_1 = \{\mu : |\mu| = 1\}$, whereas the transform $\lambda \to \beta_k$ maps the real line into itself. Clearly, powering of μ keeps the unit circle $\mathcal{C}_1$ in place, whereas the values $|\mu|^k$ converge to 0 for $|\mu| < 1$ and to $+\infty$ for $|\mu| > 1$ as $k \to \infty$; thus for large k the transform $\lambda \to \beta_k$ isolates the images of the sets of real and nonreal values λ from one another.

Corollary 10.1. *Suppose that an $n \times n$ matrix M has exactly s eigenpairs $\{\lambda_j, \mathcal{U}_j\}$, $j = 1, \dots, s$, and does not have eigenvalues $\pm\sqrt{-1}/a$. By extending the equations of Fact 10.1, write*

$$P = (aM + I_n\sqrt{-1})(aM - I_n\sqrt{-1})^{-1}, \tag{10.2}$$

$$M_k = \frac{\sqrt{-1}}{a}(P^k + 1)(P^k - 1)^{-1}, \tag{10.3}$$

$$\mu_j = (a\lambda_j + \sqrt{-1})(a\lambda_j - \sqrt{-1})^{-1},$$

$$\beta_{j,k} = \frac{n(\lambda_{j,k})}{d(\lambda_{j,k})}, \quad n(\lambda_{j,k}) = \sum_{g=0}^{\lfloor k/2 \rfloor} (-1)^g \binom{k}{2g} (a\lambda_j)^{k-2g},$$

$$d(\lambda_{j,k}) = a \sum_{g=0}^{\lfloor k/2 \rfloor} (-1)^{g+1} \binom{k}{2g+1} (a\lambda_j)^{k-2g-1},$$

$j = 1, \dots, s$. (In particular $M_1 = M$, whereas $2M_2 = M - (aM)^{-1}$.) Then $M_k = n_k(M)(d_k(M))^{-1}$ where

$$n_k(M) = \sum_{g=0}^{\lfloor k/2 \rfloor} (-1)^g \binom{k}{2g} (aM)^{k-2g},$$

$$d_k(M) = a \sum_{g=0}^{\lfloor k/2 \rfloor} (-1)^{g+1} \binom{k}{2g+1} (aM)^{h-2g-1},$$

and the matrices M_k have the eigenpairs $\{\{\beta_{j,k}, \mathcal{U}_j\},\ j = 1, \dots, s\}$ where $\beta_{j,k}$ are real if λ_j is real, $|\beta_{j,k}| + 1/|\beta_{j,k}| \to \infty$ as $k \to \infty$ unless λ_j is real.

The corollary implies that for sufficiently large integers k we can set $F(M) = M_k$ in Flowchart 5.1.

We can apply repeated squaring to compute high powers P^k. In numerical implementation we should avoid involving large norms $||P^k||_q$. Surely we can readily estimate them for $q = 1$ or $q = \infty$, but [21] proposes effective probabilistic algorithm for approximating the matrix norms $||\cdot||$. Also note that $(\rho(P))^k = \rho(P^k) \le ||P^k||_q \le ||P||_q^k$ for the spectral radii $\rho(P)$ and $\rho(P^k)$, $q = 1, 2, \infty$ and all k (cf. [72, Theorems 1.2.7 and 1.2.9]).

Below is a flowchart that implements this approach by using only two matrix inversions; this is much less than in iteration (9.5). The algorithm works for a large class of inputs M but fails for harder inputs M, which have many real and nearly real eigenvalues, but also other nonreal eigenvalues. The heuristic choice

$$v = 0,\ w = 1,\ t \approx -\Re(\text{trace}(M)),\ a = \frac{t}{n},\ \text{and } \widehat{M} = M + tI_n \tag{10.4}$$

tends to push the values $|\mu|$ away from 1 on the average input although can strongly push such a value toward 1 for the worst case input.

Flowchart 10.1. Mapping the real line onto the unit circle and repeated squaring *(cf. Remark 10.1).*

INPUT: *a real $n \times n$ matrix M, whose real and nearly real eigenvalues are associated with an unknown eigenspace $\mathcal{U}_+$ having an unknown dimension $r_+ \ll n$.*

OUTPUT: *FAILURE or a matrix $\widehat{U}$ such that $\mathcal{R}(\widehat{U}) \approx \mathcal{U}_+$.*

INITIALIZATION: *Fix sufficiently large tolerances τ and h_+, fix real a, t, v, and w and matrix $\widehat{M}$ of (10.4).*

COMPUTATIONS:

1. *Compute the matrices $P = (a\widehat{M} + I_n\sqrt{-1})(a\widehat{M} - I_n\sqrt{-1})^{-1}$ (cf. Corollary 10.1) and P^{2^g} for $g = 1, 2, \dots, h+1$ until $||P^{2^{h+1}}||_q > \tau$ for a fixed q (e.g., for $q = 1$ or $q = \infty$) or until $h \geq h_+$.*
2. *Compute matrix M_k of Corollary 10.1 for $k = 2^{h_+}$.*
3. *Apply Flowchart 5.2 to the matrix $F = M_k$ and the integer $r = n$ to output an $n \times r$ matrix basis for the strongly dominant eigenspace $\widehat{U}$ of F.*
4. *Output FAILURE if Flowchart 5.2 fails, which would mean that the matrix $F = M_k$ has no strongly dominant eigenspace of dimension $r_+ < n$.*

One can modify Stage 4 to compute an integer h_+ iteratively, according to a fixed policy: one can begin with a small h_+, then increase it and reapply the algorithm if the computations fail (see Stage 4 and see further variations in Sections 10.2 and 11).

Remark 10.1. *(a) One can extend Stage 2 by setting $N_0 = M_k$ and applying iteration (9.6). In this case cubic convergence would be exploited and we could proceed with smaller values of h_+.*

(b) In another variant one computes the matrix P^s for a sufficiently large integer s to ensure isolation of the images of real and nearly real eigenvalues of M from the images of its other eigenvalues and then applies the Rayleigh Quotient Iteration to this matrix at sufficiently many points of the unit circle $\mathcal{C}_1(0)$.

10.2 Further variations of matrix sign iteration

Let us comment on some promising variations of the matrix sign iteration.

1. We first examine how the map $M \to P$ for the matrices M and P of Corollary 10.1 transforms the basin of convergence of iteration (9.6), given by the discs $\mathcal{D}_{1/2}(\pm\sqrt{-1})$. We observe that their complement is mapped into the annulus $\mathcal{A}_{1/5,5}(0) = \{x : 1/5 \leq |x| \leq 5\}$. Conversely, suppose that under the map $M \to P^k$ the images of all nonreal eigenvalues of M lie outside this annulus, then iteration (9.6) cubically converges when it is applied to the matrix M_k of Corollary 10.1. We can estimate the desired integer k if we know the absolute values of all eigenvalues of the matrix P, that is, their distances from the origin. By virtue of part (d) of Fact 10.1 the distance is 1 if and only if an eigenvalue of P is the image of a real eigenvalue of M. Given the coefficients of the characteristic polynomial $c_P(x) = \det(xI_n - P)$, one needs $O(n \log n)$ ops to approximate all these distances with relative errors, say, at most 1% (see some effective algorithms in [70], [3], [8], [59], [61]).

2. In the case where $M = C_p$ is the companion matrix of a polynomial $p(x)$, the monic characteristic polynomial $c_P(x)$ equals $\gamma(x-1)^n p(\frac{x+1}{x-1}\frac{\sqrt{-1}}{a})$ for a scalar γ. We can compute its coefficients by using $O(n \log n)$ ops; indeed we just need to perform two shifts of the variables and the reversion of the polynomial coefficients since $\frac{x+1}{x-1} = 1 - \frac{2}{x-1}$ (see [60, Chapter 2]).

3. Having the coefficients of the characteristic polynomial $c_P(x)$ available, we can apply our algorithms to its companion matrix to compute its eigenvalues μ lying on the unit circle $\mathcal{C}_1$, and then recover the real eigenvalues $\lambda = \frac{\sqrt{-1}(\mu+1)}{a(\mu-1)}$ of M (see part (a) of Fact 10.1).

4. We can replace repeated squaring of the matrix P with k steps of the Dandelin's (Lobachevsky's, Gräffe's) root-squaring iteration [36],

$$p_{i+1}(x) = (-1)^n p_i(\sqrt{x}) p_i(\sqrt{-x}), \quad i = 0, 1, \dots, k-1 \tag{10.5}$$

for $p_0(x) = c_P(x)$. We have $p_i(x) = \prod_{j=1}^{n}(x - \lambda_j^{2^i})$, so that the ith iteration step squares the roots of the polynomial $p_{i-1}(x)$ for every i. Every root-squaring step (10.5) essentially amounts to polynomial multiplication and can be performed in $O(n \log n)$ ops; one can improve numerical stability by increasing this count to order n^2 [52]. Having computed the polynomial $p_k(x)$, for a sufficiently large integer k, we have its roots on the unit circle sufficiently well isolated from its other roots. The application of the algorithm in the next section to C_{p_k}, the companion matrix of this polynomial, yields its roots lying on $\mathcal{C}_1$ (they are the eigenvalues of C_{p_k}). From these roots we can recover the roots μ of the circle $c_P(x) = p_0(x)$ by means of the descending techniques of [54] (applied also in [55], [56], [61], and [66, Stage 8 of Algorithm 9.1]), and then can recover the real roots λ of $p(x)$ from the values μ by applying the expression in part (a) of Fact 10.1.

Remark 10.2. *Having isolated the roots of $p_k(x)$ on the circle $\mathcal{C}_1$ from its other roots, we can apply the algorithms of [70], [54], [55], [45], [61] to split out the factor $f(x)$ of this polynomial sharing with $p_k(x)$ precisely all the roots on the circle $\mathcal{C}_1$. Then these roots can be readily approximated based on the Laguerre or modified Laguerre algorithms. Numerical problems can be caused by potentially dramatic growth of the coefficients of $p_k(x)$ in the transition to the factor $f(x)$ unless its degree is small.*

11 Repeated squaring and the Möbius transform

Our next iteration begins as Flowchart 10.1, but we interrupt repeated squaring by applying the scaled Möbius transform $x \to x + 1/x$, instead of the map $P \to M_k$ of (10.3). The scaled Möbius transform moves the images of all real eigenvalues of the matrix M from the unit circle $\mathcal{C}_1$ into the real line interval $[-2, 2]$; furthermore under this transform of the matrix N_i the images of all its eigenvalues lying outside the annulus $\mathcal{A}_{1/3,3}(0) = \{x : \ 1/3 \le |x| \le 3\}$ are moved into the exterior of the disc $D_{8/3}(0)$. Recall that the basin of convergence of iteration (9.6) preceded by the map $x \to \frac{\sqrt{-1}}{a}\frac{1+x}{1-x}$ was the exterior of the slightly larger annulus $\mathcal{A}_{1/5,5}(0) = \{x : \ 1/5 \le |x| \le 5\}$; furthermore the Möbius transform numerically stabilizes the computations for a large class of inputs.

Next we comment on combining the maps of Fact 10.1, repeated squaring, and the Möbius transform; we observe some pitfalls and propose remedies.

Fact 11.1. *(Cf. Fact 10.1 for $a = 1$.) Write*

$$\mu = (\lambda + \sqrt{-1})(\lambda - \sqrt{-1})^{-1}. \tag{11.1}$$

Then

(a) $\lambda = \sqrt{-1}(\mu - 1)/(\mu + 1)$,

(b) $|\mu| = 1$ if and only if λ is real and

(c) $\mu_k = \mu^k + \mu^{-k} = \sum_{g=0}^{k}(-1)^g \binom{2k}{2g} \lambda^{2k-2g}(\lambda^2+1)^{-k}$ for $k = 1, 2, \dots$. (In particular $\mu_1 = \frac{\lambda^2-1}{\lambda^2+1}$, whereas $\mu_2 = \frac{\lambda^4-6\lambda^2+1}{(\lambda^2+1)^2}$.)

Fact 11.2. *Assume μ of (11.1) and a nonnegative integer k. Then $|\mu| = 1$ and $-2 \le \mu_k = \mu^k + \mu^{-k} \le 2$ if λ is real, whereas $|\mu^k + \mu^{-k}| \to \infty$ as $k \to \infty$ otherwise.*

Corollary 11.1. *Let an $n \times n$ matrix M have exactly s eigenpairs $\{\lambda_j, \mathcal{U}_j\}$, $j = 1, \dots, s$, and not have eigenvalues $\pm\sqrt{-1}$. By extending (10.2) for $a = 1$ and (11.1), write*

$$P = (M + I_n\sqrt{-1})(M - I_n\sqrt{-1})^{-1} = (M - I_n\sqrt{-1})^{-1}(M + I_n\sqrt{-1}),$$

$$T_k = P^k + P^{-k} = \sum_{g=0}^{k} (-1)^g \binom{2k}{2g} M^{k-2g}(M^2+1)^{-k}, \tag{11.2}$$

$$\mu_j = (\lambda_j + \sqrt{-1})(\lambda_j - \sqrt{-1})^{-1},$$

$$\mu_{j,k} = \mu_j^k + \mu_j^{-k} = \sum_{g=0}^{k} (-1)^g \binom{2k}{2g} \lambda_j^{k-2g}(\lambda_j^2+1)^{-k}$$

for $k = 1, 2, \dots$ (In particular $T_1 = 2(I_n - M^2)(I_n + M^2)^{-1} = 2I_n - 4(I_n + M^2)^{-1}$, whereas $T_2 = (M^4 - 6M^2 + I_n)(M^2 + I_n)^{-2} = (M^2 + I_n)^{-2}(M^4 - 6M^2 + I_n)$.) Then $M = \sqrt{-1}(P - I_n)(P + I_n)^{-1} = \sqrt{-1}(P + I_n)^{-1}(P - I_n)$, $\lambda_j = \sqrt{-1}(\mu_j - 1)/(\mu_j + 1)$ for $j = 1, \dots, s$, and the matrices T_k have the eigenpairs $\{\{\mu_{j,k}, \mathcal{U}_j\},\ j = 1, \dots, s\}$ where $-2 \le \mu_{j,k} \le 2$ if λ_j is real, $|\mu_{j,k}| \to \infty$ as $h \to \infty$ unless λ_j is real.

Instead of the map $P^k \to M_k$ and equation (10.3) of Corollary 10.1 we employ the map $P^k \to T_k$ and equation (11.2). This complicates the isolation of the images of real eigenvalues of the matrix M from the images of its nonreal eigenvalues provided that we rely on the respective map of the eigenvalues $\lambda = \lambda(P^k) \to \lambda(T_k) = \lambda + 1/\lambda$. Indeed the unit circle $\{\lambda = \lambda(P^k) : |\lambda| = 1\}$ is still mapped onto the line segment $[-2, 2]$, but also the imaginary line $\{\lambda = \lambda(P^k) : \Re(\lambda) = 0\}$ is mapped into the real line.

The problem disappears, however, where $\max\{|\lambda|, 1/|\lambda|\} > 3$, because in this domain the value $|\lambda + 1/\lambda|$ exceeds $8/3$, whereas this value is small near the points $\lambda = \pm\sqrt{-1}$. Therefore we can safely apply the map $P^k \to T_k$ provided that the images of nonreal eigenvalues of M in the map $M \to P^k$ do not lie in the annulus $\mathcal{A}_{1/3,3}(0) = \{x :\ 1/3 \le |x| \le 3\}$.

This map does not enable stabilization, because it does not generally increase the minimum ratio of the absolute values of the images of real and nonreal eigenvalues of M, but it brings the images of all nonreal eigenvalues of M into the exterior of the disc $D_{8/3}$, while sending the images of all real eigenvalues of M into the real line interval $[-2, 2]$.

If at this stage we can afford a reasonably large number of squarings of the resulting matrix (resp. its inverse), then the eigenspace associated with real eigenvalues of M becomes dominated (resp. dominant), and we can approximate them by applying Flowchart 5.3 (resp. 5.1).

12 Numerical tests

We performed a series of numerical tests in the Graduate Center of the City University of New York using a Dell server with a dual core 1.86 GHz Xeon processor and 2G memory running Windows Server 2003 R2. The test Fortran code was compiled with the GNU gfortran compiler within the Cygwin environment. We generated random numbers with the random_number intrinsic Fortran function assuming the uniform probability distribution over the range $\{x :\ 0 \le x < 1\}$. To shift to the range $\{y :\ b \le y \le a + b\}$ for fixed real a and b, we applied the linear transform $x \to y = ax + b$.

Conditioning of the products with random Toeplitz matrices.

Table 12.1 displays the average residual norms $rn = ||ATY - T_{q,A}||$ where A is an $n \times n$ matrix having numerical rank q, $Y = (AT)^+ T_{q,A}$, $\mathcal{R}(T_{q,A}) = \mathbb{T}_{q,A}$ is the leading singular space of A, and T is a random $n \times q$ Toeplitz matrix. We performed 100 tests for each pair $\{n, q\}$ for $n = 64, 128, 256$ and $q = 8, 32$.

We have first generated the Q factors S and T of $n \times n$ random matrices as well as the diagonal matrices $\Sigma = \mathrm{diag}(\sigma_j)_{j=1}^n$ such that $\sigma_j = 1/j,\ j = 1, \dots, q$, $\sigma_j = 10^{-10},\ j = q+1, \dots, n$, $||A|| = 1$, $\kappa(A) = ||A^{-1}|| = 10^{10}$. Then we computed the input matrices $A = S\Sigma T^T$. The average residuals norms rn had the same order in our tests with random $n \times q$ general multipliers M, replacing the Toeplitz multipliers T.

Algorithms and tests

Table 12.1: Residual norms rn with random $n \times q$ Toeplitz multipliers T.

q	n	**min**	**max**	**mean**	**std**
8	64	2.22×10^{-09}	7.89×10^{-06}	1.43×10^{-07}	9.17×10^{-07}
8	128	3.79×10^{-09}	4.39×10^{-05}	4.87×10^{-07}	4.39×10^{-06}
8	256	5.33×10^{-09}	3.06×10^{-06}	6.65×10^{-08}	3.12×10^{-07}
32	64	6.22×10^{-09}	5.00×10^{-07}	4.06×10^{-08}	6.04×10^{-08}
32	128	2.73×10^{-08}	4.88×10^{-06}	2.57×10^{-07}	8.16×10^{-07}
32	256	1.78×10^{-08}	1.25×10^{-06}	1.18×10^{-07}	2.03×10^{-07}

We tested our algorithms for the approximation of the eigenvalues of $n \times n$ companion matrix C_p and of the shifted matrix $C_p - sI_n$ defined by polynomials $p(x)$ with random real coefficients for $n = 64, 128, 256$ and by random real s. For each class of matrices, each input size and each iterative algorithm we generated 100 input instances and run 100 tests. Our tables show the minimum, maximum, and average (mean) numbers of iteration loops in these runs (until convergence) as well as the standard deviations in the columns marked by "**min**", "**max**", "**mean**", and "**std**", respectively.

We applied repeated squaring of Section 6 to the matrix $C_p - sI$; we used shifts s because polynomials $p(x)$ with random real coefficients tend to have all roots near the circle $\mathcal{C}_1(0)$ and because for such inputs repeated squaring of C_p advances eigen-solving very slowly.

We applied real Newton's iteration (9.5) to approximate the matrix sign function for the matrix C_p using no shifts; then we applied Flowchart 5.1 to approximate real eigenvalues.

In both groups of tests above we output roots with at least four correct decimals. In our next group of tests we output roots with at least three correct decimals. In these tests we applied real Padé iteration (9.6) without stabilization to the matrices produced by five Newton's steps (9.5).

Table 12.2 displays the results of testing repeated squaring of Section 6. The first three lines show the dimension of the output subspace and the matrix L. The next three lines show the number of squarings performed until convergence.

Table 12.3 displays the number of Newton's steps (9.5) performed until convergrence.

Table 12.4 covers the tests where we first performed five Newton's steps (9.5) followed by sufficiently many Padé steps (9.6) required for convergence. The first three lines of the table show the number of the Padé steps. The next three lines display the percent of the real roots of the polynomials $p(x)$ which the algorithm computed with at least three correct decimals (compared to the overall number of the real eigenvalues of L). The next three lines show the increased percent of computed roots when we refined the crude approximations by means of Rayleigh Quotient iteration. The iteration rapidly converged from all these initial approximations but in many cases to the same roots from distinct initial points.

Table 12.5 shows the increased percent of computed roots where we applied our algorithms to both polynomials $p(x)$ and $p_{\rm rev}(x)$.

Table 12.2: Repeated Squaring

n	**dimension/squarings**	**min**	**max**	**mean**	**std**
64	dimension	1	10	5.31	2.79
128	dimension	1	10	3.69	2.51
256	dimension	1	10	4.25	2.67
64	squarings	6	10	7.33	0.83
128	squarings	5	10	7.37	1.16
256	squarings	5	11	7.13	1.17

Table 12.3: Newton's iteration (9.5).

n	**min**	**max**	**mean**	**std**
64	7	11	8.25	0.89
128	8	11	9.30	0.98
256	9	13	10.22	0.88

Table 12.4: 5 N-steps (9.5) + P-steps (9.6)

n	**P-steps or %**	**min**	**max**	**mean**	**std**
64	P-steps	1	4	2.17	0.67
128	P-steps	1	4	2.05	0.63
256	P-steps	1	3	1.99	0.58
64	% w/o RQ steps	0	100	64	28
128	% w/o RQ steps	0	100	39	24
256	% w/o RQ steps	0	100	35	20
64	% w/RQ steps	0	100	89	19
128	% w/RQ steps	0	100	74	26
256	% w/RQ steps	0	100	75	24

References

[1] E. T. Bell, *The Development of Mathematics*, McGraw-Hill, New York, 1940.

[2] C. A. Boyer, *A History of Mathematics*, Wiley, New York, 1968.

[3] D. A. Bini, Numerical Computation of Polynomial Zeros by Means of Aberth's Method, *Numerical Algorithms*, **13**, 179–200, 1996.

[4] R. Bevilacqua, E. Bozzo, G. M. Del Corso, Qd-type Methods for Quasiseparable Matrices, SIAM J. on Matrix Analysis and Applications, **32**, **3**, 722747, 2011.

[5] D. A. Bini, P. Boito, Y. Eidelman, L. Gemignani, I. Gohberg, A Fast Implicit QR Algorithm for Companion Matrices, *Linear Algebra and Its Applications*, **432**, 2006–2031, 2010.

[6] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, H. van der Vorst, editors, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.

[7] D. A. Bini, F. Daddi, and L. Gemignani, On the shifted QR iteration applied to companion matrices, *Electron. Transactions on Numerical Analysis (ETNA)*, **18**, 137–152, 2004.

[8] D. A. Bini, G. Fiorentino, Design, Analysis, and Implementation of a Multiprecision Polynomial Rootfinder, *Numerical Algorithms*, **23**, 127–173, 2000.

[9] A. Böttcher, S. M. Grudsky, *Spectral Properties of Banded Toeplitz Matrices*, SIAM Publications, Philadelphia, 2005.

[10] D. A. Bini, L. Gemignani, V. Y. Pan, Inverse Power and Durand/Kerner Iteration for Univariate Polynomial Root-finding, *Computers and Mathematics (with Applications)*, **47**, **2/3**, 447–459, 2004. (Also Technical Report TR 2002 020, *CUNY Ph.D. Program in Computer Science, Graduate Center, City University of New York*, 2002.)

[11] D. A. Bini, L. Gemignani, V. Y. Pan, Algorithms for Generalized Companion Matrices and Secular Equation, *Numerische Math.* **3**, 373–408, 2005. Also Technical Report 1470, *Department of Math., University of Pisa*, Pisa, Italy (July 2003).

[12] D. A. Bini, L. Gemignani, V. Y. Pan, Improved Initialization of the Accelerated and Robust QR-like Polynomial Root-finding, *Electronic Transactions on Numerical Analysis* **17**, 195–205, 2004. Proc. version in CASC'2004.

[13] D. Bini, V. Y. Pan, Parallel Complexity of Tridiagonal Symmetric Eigenvalue Problem, *in Proceedings of 2nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'91)*, 384-393, ACM Press, New York, and SIAM Publications, Philadelphia, January 1991.

[14] D. Bini, V. Y. Pan, Practical Improvement of the Divide-and-Conquer Eigenvalue Algorithms, *Computing*, **48**, 109-123 (1992).

[15] D. Bini, V. Y. Pan, Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real, *SIAM Journal on Computing*, **27**, **4**, 1099–1115, 1998.

[16] D. Bini, V. Y. Pan, Graeffe's, Chebyshev, and Cardinal's Processes for Splitting a Polynomial into Factors, *J. Complexity*, **12**, 492–511, 1996.

[17] M. Ben-Or, P. Tiwari, Simple Algorithms for Approximating All Roots of a Polynomial with Real Roots, *J. of Complexity*, **6, 4**, 417–442, 1996.

[18] J. P. Cardinal, On Two Iterative Methods for Approximating the Roots of a Polynomial, *Lectures in Applied Mathematics*, **32** (*Proceedings of AMS-SIAM Summer Seminar: Mathematics of Numerical Analysis: Real Number Algorithms* (J. Renegar, M. Shub, and S. Smale, editors), Park City, Utah, 1995), 165–188, American Mathematical Society, Providence, Rhode Island, 1996.

[19] F. Cajori, *A History of Mathematics, 5/E*, AMS Chelsea Publ., Providence, Rhode Island, 1999.

[20] Z. Chen, J. J. Dongarra, Condition Numbers of Gaussian Random Matrices, *SIAM. J. on Matrix Analysis and Applications*, **27**, 603–620, 2005.

[21] J. D. Dixon, Estimating Extremal Eigenvalues and Condition Numbers of Matrices, *SIAM J. on Numerical Analysis*, **20, 4**, 812–814, 1983.

[22] J. Demmel, The Probability That a Numerical Analysis Problem Is Difficult, *Math. of Computation*, **50**, 449–480, 1988.

[23] Q. Du, M. Jin, T. Y. Li, Z. Zeng, Quasi-Laguerre Iteration in Solving Symmetric Tridiagonal Eigenvalue Problems, *SIAM J. Sci. Computing*, **17**, **6**, 1347–1368, 1996.

[24] Q. Du, M. Jin, T. Y. Li, Z. Zeng, The Quasi-Laguerre Iteration, *Math. of Computation*, **66**, **217**, 345–361, 1997.

[25] K. R. Davidson, S. J. Szarek, Local Operator Theory, Random Matrices, and Banach Spaces, in *Handbook on the Geometry of Banach Spaces* (W. B. Johnson and J. Lindenstrauss editors), pages 317–368, North Holland, Amsterdam, 2001.

[26] A. Edelman, Eigenvalues and Condition Numbers of Random Matrices, *SIAM J. on Matrix Analysis and Applications*, **9**, **4**, 543–560, 1988.

[27] A. Edelman, B. D. Sutton, Tails of Condition Number Distributions, *SIAM J. on Matrix Analysis and Applications*, **27**, **2**, 547–560, 1988.

[28] A. Eigenwillig, V. Sharma, C. K. Yap, Almost Tight Recursion Tree Bounds for the Descartes Method, *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC 2006)*, 71–78, ACM Press, New York, 2006.

[29] I. Z. Emiris, B. Mourrain, E. Tsigaridas, Real Algebraic Numbers: Complexity Analysis and Experimentation, in *RELIABLE IMPLEMENTATIONS OF REAL NUMBER ALGORITHMS: THEORY AND PRACTICE, LNCS*, **5045**, 57–82, Springer, 2008 (also available in www.inria.fr/rrrt/rr-5897.html).

[30] A. Galligo, M. E. Alonso, A Root Isolation Algorithm for Sparse Univariate Polynomials, *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC 2012)*, ACM Press, New York, 2012.

[31] I. Gohberg, N. Y. Krupnick, A Formula for the Inversion of Finite Toeplitz Matrices, *Matematicheskiie Issledovaniia* (in Russian), **7**, **2**, 272–283, 1972.

[32] G. H. Golub, C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1996 (third edition).

[33] S. A. Goreinov, I. V. Oseledets, D. V. Savostyanov, E. E. Tyrtyshnikov, N. L. Zamarashkin, How to Find a Good Submatrix, Research Report 08-10, ICM HKBU, Kowloon Tong, Hong Kong, 2008.

[34] S. A. Goreinov, E. E. Tyrtyshnikov, The Maximal-volume Concept in Approximation by Low-rank Matrices, *Contemporary Mathematics*, **208**, 47–51, 2001.

[35] S. A. Goreinov, E. E. Tyrtyshnikov, N. L. Zamarashkin, A Theory of Pseudo-skeleton Approximations, *Linear Algebra and Its Applications*, **261**, 1–22, 1997.

[36] A. S. Householder, Dandelin, Lobachevskii, or Graeffe, *American Mathematical Monthly* **66**, 464–466, 1959.

[37] A. S. Householder, Generalization of an algorithm by Sebastiao e Silva, *Numerische Math.,*, **16**, 375–382, 1971.

[38] N. J. Higham, *Functions of Matrices: Theory and Computations*, SIAM, Philadelphia, 2008.

[39] N. Halko, P. G. Martinsson, J. A. Tropp, Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions, *SIAM Review*, **53, 2**, 217–288, 2011.

[40] Y. P. Hong, C.-T. Pan, Rank-Revealing QR Factorizations and the Singular Value Decomposition, *Mathematics of Computation*, **58**, **197**, 213-232, 1992.

[41] E. Hansen, M. Patrick, J. Rusnack, Some Modification of Laguerre's Method, *BIT*, **17**, 409–417, 1977.

[42] J. Hubbard, D. Schleicher, S. Sutherland, How to Find All Roots of Complex Polynomials by Newton's Method, *Invent. Math.*, **146**, 1–33, 2001.

[43] M. Hemmer, E.P. Tsigaridas, Z. Zafeirakopoulos, I. Z. Emiris, M. I. Karavelas, B. Mourrain, Experimental Evaluation and Cross-Benchmarking of Univariate Real Solvers, in *Proc. International Symposium on Symbolic-Numerical Computations*, (Kyoto, Japan, August 2009), (edited by Hiroshi Kai and Hiroshi Sekigawa), pp.105–113, ACM Press, New York, 2009.

[44] R. J. Johnston, Gershgorin Theorems for Partitioned Matrices, *Linear Algebra and Its Applications*, **4**, 205–220, 1971.

[45] P. Kirrinnis, Polynomial Factorization and Partial Fraction Decomposition by Simultaneous Newton's Iteration, *J. of Complexity*, **14**, 378–444, 1998.

[46] J.M. McNamee, A 2002 update of the supplementary bibliography on root of polynomials, *J. of Computational and Applied Math.* **142**, 433-434, 2002; also at web-site `www.yorku.ca/~mcnamee/`

[47] J.M. McNamee, *Numerical Methods for Roots of Polynomials (Part 1)*, Elsevier, Amsterdam, 2007.

[48] J.M. McNamee and V.Y. Pan, Efficient polynomial root-refiners: a survey and new record estimates, *Computers and Math. (with Applications)*, **63**, 239–254, 2012.

[49] J.M. McNamee and V.Y. Pan, *Numerical Methods for Roots of Polynomials, Part 2*, 780+XIX pages, submitted to Elsevier publishers.

[50] K. Mehlhorn, M. Sagraloff, A Deterministic Algorithm for Isolating Real Roots of a Real Polynomial, *J. of Symbolic Computation* **46, 1**, 70–90, 2011.

[51] H. I. Medley, K. S. Varga, On Smallest Isolated Gerschgorin Disks for Eigenvalues, *Numerische Mathematik*, **11**, 361–369, 1968.

[52] G. Malajovich, J. P. Zubelli, On the Geometry of Graeffe Iteration, *J. of Complexity*, **17, 3**, 541-573, 2001.

[53] B. Parlett, Laguerre's Method Applied to the Matrix Eigenvalue Problem, *Math. of Computation*, **18**, 464–485, 1964.

[54] V. Y. Pan, Optimal (up to polylog factors) sequential and parallel algorithms for approximating complex polynomial zeros, *Proc. 27th Ann. ACM Symp. on Theory of Computing*, 741–750, ACM Press, New York, 1995.

[55] V. Y. Pan, Optimal and nearly optimal algorithms for approximating polynomial zeros, *Computers and Math. (with Applications)* **31, 12**, 97–138, 1996.

[56] V. Y. Pan, Solving a Polynomial Equation: Some History and Recent Progress, *SIAM Review*, **39, 2**, 187–220, 1997.

[57] V. Y. Pan, Solving Polynomials with Computers, *American Scientist*, **86**, January–February 1998. Available via

http://comet.lehman.cuny.edu/vpan/research/publications

[58] C.–T. Pan, On the Existence and Computation of Rank-revealing LU Factorization, *Linear Algebra and Its Applications*, **316**, 199–222, 2000.

[59] V. Y. Pan, Approximating Complex Polynomial Zeros: Modified Quadtree (Weyl's) Construction and Improved Newton's Iteration, *J. of Complexity*, **16**, **1**, 213–264, 2000.

[60] V. Y. Pan, *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser, Boston, and Springer, New York, 2001.

[61] V. Y. Pan, Univariate Polynomials: Nearly Optimal Algorithms for Factorization and Rootfinding, *Journal of Symbolic Computations*, **33, 5**, 701–733, 2002. Proc. version in *Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC 01),* 253–267, ACM Press, New York, 2001.

[62] V. Y. Pan, Amended DSeSC Power Method for Polynomial Root-finding, *Computers and Math. (with Applications)*, **49**, **9**–**10**, 1515–1524, 2005.

[63] V. Y. Pan, Univariate Polynomial Root-Finding by Arming with Constraints, *Proc. of the Forth International Symposium on Symbolic-Numerical Computations (SNC '2011)*, San Jose, California, June 2011 (edited by Marc Moreno Maza), 112–121, ACM Press, New York, 2011.

[64] V. Y. Pan, G. Qian, A. Zheng, Randomized Preconditioning of the MBA Algorithm, *Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC 2011)*, (San Jose, California, June 2011), (edited by Anton Leykin), 281–288, ACM Press, New York, 2011.

[65] V. Y. Pan, G. Qian, A. Zheng, Randomized and Derandomized Matrix Computations, Tech. Report TR 2011011, *Ph.D. Program in Computer Science*, *Graduate Center, the City University of New York*, 2011.

[66] V. Y. Pan, A. Zheng, New Progress in Real and Complex Polynomial Root-Finding, *Computers and Math. (with Applications)* **61**, 1305–1334. Proceedings version: Real and Complex Polynomial Root-Finding with Eigen-Solving and Preprocessing, in *Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC 2010)*, pages 219–226, ACM Press, New York, 2010.

[67] V. Y. Pan, A. Zheng, Root-Finding by Expansion with Independent Constraints, *Computers and Math. (with Applications)* **62**, 3164-3182, 2011.

[68] J. Sebastiao e Silva, Sur une Méthode d'Approximation Semblable a Celle de Graeffe, *Portugal Math.*, **2**, 271–279, 1941.

[69] G. W. Stewart, On the Convergence of Sebastiao E Silva's Method for Finding a Zero of a Polynomial, *SIAM Review*, **12**, 458–460, 1970.

[70] A. Schönhage, The Fundamental Theorem of Algebra in Terms of Computational Complexity, *Mathematics Department, University of Tübingen*, Germany, 1982.

[71] G. W. Stewart, *Matrix Algorithms, Vol I: Basic Decompositions*, SIAM, Philadelphia, 1998.

[72] G. W. Stewart, *Matrix Algorithms, Vol II: Eigensystems*, SIAM, Philadelphia, 2001 (second edition).

[73] M. Sagraloff, When Newton meets Descartes: A Simple and Fast Algorithm to Isolate the Real Roots of a Polynomial, *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC 2012)*, ACM Press, New York, 2012.

[74] A. Sankar, D. Spielman, S.-H. Teng, Smoothed Analysis of the Condition Numbers and Growth Factors of Matrices, *SIAM Journal on Matrix Analysis*, **28**, **2**, 446–476, 2006.

[75] A. Strzebonski, E. Tsigaridas, Univariate Real Root Isolation in Multiple Extension Fields, *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC 2012)*, ACM Press, New York, 2012.

[76] V. Sharma, C. Yap, Near Optimal Tree Size Bounds on a Simple Real Root Isolation Algorithm, *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC 2012)*, ACM Press, New York, 2012.

[77] E. P. Tsigaridas, I. Z. Emiris, Univariate polynomial real root isolation: Continued Fractions revisited, *ESA'06 Proceedings of the 14th Conference on Annual European Symposium*, Zurich, 2006, *LNCS*, **4168**, 817–828, Springer, London, 2006.

[78] R. S. Varga, Minimal Gerschgorin sets for partitioned matrices, *SIAM J. on Numericaql Analysis*, **7**, 493–507, 1970.

[79] M. Van Barel, R. Vandebril, P. Van Dooren, K. Frederix, Implicit Double Shift QR-algorithm for Companion Matrices, *Numerische Mathematik* **116**, **2**, 177–212, 2010.

[80] D. S. Watkins, *Fundamentals of Matrix Computations*, Wiley, New York, 2002 (second edition).

[81] D. S. Watkins, *The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods*, SIAM, Philadelphia, PA, 2007.

[82] C. Yap, M. Sagraloff, A Simple but Exact and Efficient Algorithm for Complex Root Isolation, *Proc. of International Symp. on Symbolic and Algebraic Computation (ISSAC '11)*, San Jose, California, June 2011 (edited by A. Leykin), 353–360, ACM Press, New York, 2011.

[83] X. Zou, Analysis of the Quasi-Laguerre Method, *Numerische Math.*, **82**, 491–519, 1999.