

Finding the Distance to Instability of a Large Sparse Matrix

Daniel Kressner

Abstract—The distance to instability of a matrix A is a robust measure for the stability of the corresponding dynamical system $\dot{x} = Ax$, known to be far more reliable than checking the eigenvalues of A . In this paper, a new algorithm for computing such a distance is sketched. Built on existing approaches, its computationally most expensive part involves a usually modest number of shift-and-invert Arnoldi iterations. This makes it possible to address large sparse matrices, such as those arising from discretized partial differential equations.

I. INTRODUCTION

Let the $n \times n$ matrix A be stable, that is, its set of eigenvalues $\Lambda(A)$ is contained in the open left half plane. Then the *distance to instability* of A is defined as

$$\mu(A) := \inf\{\|E\| : E \in \mathbb{C}^{n \times n}, A + E \text{ is not stable}\}, \quad (1)$$

where $\|A\|$ denotes the matrix 2-norm.

If the dynamical system $\dot{x} = Ax$ represents the approximation to a nonlinear or infinite-dimensional system then the eigenvalues of A may give misleading information about the stability of the original system; an aspect that is particularly emphasized in [32]. Even when $\dot{x} = Ax$ can be considered to be exact, the eigenvalues tell little about the transient behavior of the system if A happens to be highly nonnormal [19]. In such cases, the distance to instability provides a more robust measure of stability and might be easier to grasp than pseudospectra or similar concepts that take the nonnormality of A into account. Another application is that $\mu(A)$ gives an upper bound on the smallest singular value of the Lyapunov operator $X \mapsto AX + XA^T$ [33], which in turn can be used to estimate the sensitivity of Lyapunov equations.

Finding $\mu(A)$ for large n represents a computational challenge. All known algorithms capable of determining $\mu(A)$ precisely, rely, to the best of our knowledge, on computing purely imaginary eigenvalues of $2n \times 2n$ Hamiltonian matrices. So far, dense (structure-preserving) eigenvalue solvers [5], [34], [4] have been proposed to address this task, requiring $O(n^2)$ memory and $O(n^3)$ computing time. Consequently, such an approach is limited to medium-sized matrices, say $n = O(10^4)$. While much attention has been paid to determine the rightmost eigenvalues of a large sparse matrix effectively, see [25], [28], [27], surprisingly few attempts [16] have been made to compute the distance to instability without having to resort to dense linear algebra techniques.

Supported by the DFG Emmy Noether fellowship KR 2950/1-1 and by the Swedish Foundation for Strategic Research under grant A3 02:128.

D. Kressner is with Department of Computing Science and HPC2N, Umeå University, S-901 87 Umeå, Sweden kressner@cs.umu.se

In this short paper, we sketch a new approach which combines ideas from [6], [15], [16] to obtain an algorithm capable of dealing with a large sparse matrix A , provided that the sparsity pattern of A admits the fast solution of linear systems involving the associated Hamiltonian matrices. One of the key ingredients is the use of a structure-preserving inverse iteration/Arnoldi method [29], [22] to find the (presumably few) purely imaginary eigenvalues of a Hamiltonian matrix in a reliable manner. (A similar approach has been described in [15] for locating the real eigenvalues of a large matrix.)

The rest of this paper is organized as follows. In Section II, existing approaches to computing $\mu(A)$ are briefly described. Section III is concerned with the closely related task of finding all purely imaginary eigenvalues of a Hamiltonian matrix. To make this task computationally feasible it is important to provide a good upper bound on $\mu(A)$, which is discussed in Section IV. Some preliminary numerical experiments, demonstrating that our new approach computes $\mu(A)$ for large matrices, are presented in Section V.

II. EXISTING APPROACHES

The continuity of eigenvalues implies that the infimum in (1) is attained for perturbations moving eigenvalues of A to the imaginary axis. In turn,

$$\mu(A) = \min_{\omega \in \mathbb{R}} \sigma_{\min}(A - i\omega I), \quad (2)$$

where $\sigma_{\min}(\cdot)$ denotes the minimum singular value of a matrix, see also [33]. This suggests the application of a general optimization method to (2). One difficulty with such an approach is that the function $\sigma_{\min}(A - i\omega I)$ is generally not convex and can have as much as $2n$ stationary points [8]. To check for global minima, the following result is of fundamental importance.

Theorem 1 ([9]): The Hamiltonian matrix

$$H(\epsilon) = \begin{bmatrix} A & -\epsilon I \\ \epsilon I & -A^* \end{bmatrix} \quad (3)$$

has an eigenvalue on the imaginary axis if and only if $\epsilon \geq \mu(A)$.

A. Byers' bisection method

Theorem 1 suggests a simple bisection strategy to compute $\mu(A)$ [9], [18]. Starting with a lower bound l and an upper bound u , the eigenvalues of $H(\epsilon)$ evaluated at the geometric mean $\epsilon = \sqrt{lu}$ are computed. If $H(\epsilon)$ has eigenvalues on the imaginary axis, Theorem 1 implies $\epsilon \geq \mu(A)$ and the upper bound u can therefore be replaced by $u = \epsilon$. Otherwise, $\epsilon < \mu(A)$ and the new lower bound is set to $l = \epsilon$.

The described algorithm converges linearly to $\mu(A)$. Deciding whether $H(\epsilon)$ has eigenvalues on the imaginary axis in finite-precision arithmetic can be implemented in a safe manner, provided that a structure-preserving eigenvalue solver is used for determining $\Lambda(H(\epsilon))$ [9], [33]. In this case, $\mu(A)$ can be computed to the level of roundoff error (although such a goal is rather expensive due to the slow convergence of the bisection method).

B. The Boyd-Balakrishnan method

In [6] and [7], a quadratically convergent method has been proposed to address the more general task of computing the H_∞ norm of a transfer function matrix. To illustrate this approach for the special case of computing $\mu(A)$, let us consider the following matrix [13]:

$$A = \begin{bmatrix} -1 & -5 & -25 & -125 & -625 \\ 0 & -1 & -5 & -25 & -125 \\ 0 & 0 & -1 & -5 & -25 \\ 0 & 0 & 0 & -1 & -5 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}, \quad (4)$$

for which $\mu(A) \approx 0.008$.

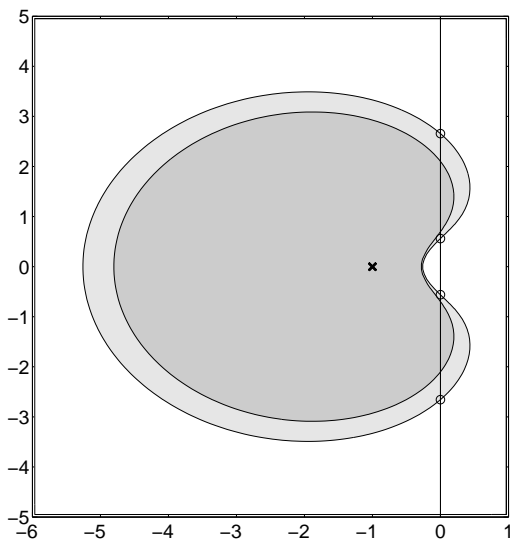


Fig. 1. Pseudospectra of the Demmel matrix (4) for $\epsilon = 0.015$ (outer area) and $\epsilon = 0.0112$ (inner are).

Figure 1 displays the pseudospectrum of A , defined as

$$\Lambda_\epsilon(A) := \bigcup_{\substack{E \in \mathbb{C}^{n \times n} \\ \|E\| \leq \epsilon}} \Lambda(A + E),$$

for $\epsilon = 0.015$. It turns out that the imaginary axis crosses the boundary of the pseudospectrum (marked by circles in Figure 1). Therefore it follows from (1) that $\mu(A) \leq \epsilon$. Moreover, the crossing points iy_1, \dots, iy_4 , ordered so that $y_1 \leq \dots \leq y_4$, are given by the purely imaginary eigenvalues of the Hamiltonian matrix $H(\epsilon)$ defined in (3), see, e.g., [8]. Setting

$$z_1 = \frac{y_1 + y_2}{2}, \quad z_2 = \frac{y_3 + y_4}{2},$$

it follows that $\sigma_{\min}(A - iz_j I) < \epsilon$ for $j = 1, 2$, since both iz_1 and iz_2 are in the interior of the pseudospectrum. Hence, we obtain a smaller ϵ satisfying $\epsilon \geq \mu(A)$ by updating

$$\epsilon = \min\{\sigma_{\min}(A - iz_1 I), \sigma_{\min}(A - iz_2 I)\}.$$

In our example, the new ϵ is given by 0.0112 which yields the smaller pseudospectrum shown in Figure 1. The described procedure is repeated with this updated value to eventually obtain $\epsilon \rightarrow \mu(A)$.

For general matrices, we have the following algorithm.

Algorithm 2 (Boyd-Balakrishnan method):

Input: $A \in \mathbb{C}^{n \times n}$ and $\epsilon > 0$ with $\epsilon \geq \mu(A)$.
Output: Updated upper bound ϵ such that $\mu(A) \approx \epsilon$.

while not converged **do**

- 1 Compute purely imaginary eigenvalues $iy_1, iy_2, \dots, iy_{2m}$ of $H(\epsilon)$ defined in (3), ordered so that $y_1 \leq \dots \leq y_{2m}$.
- 2 Update

$$\epsilon = \min_{j=1,3,\dots,2m-1} \left\{ \sigma_{\min} \left(A - i \frac{y_{j+1} - y_j}{2} I \right) \right\}.$$

end while

Algorithm 2 can be shown to exhibit global and quadratic local convergence. The use of structure-preserving eigenvalue solvers in Step 1, such as those described in [5], [34], [4], limits the effect of roundoff errors on purely imaginary eigenvalues and improves the overall reliability of the algorithm, see also [24] for more details. Efficient methods for computing/estimating the smallest singular values of the parameter-dependent matrices in Step 2 can be found, e.g., in [33].

C. Other methods

A variant of Algorithm 2, which takes the eigenvector information of $H(\epsilon)$ into account to accelerate the local convergence rate, is described in [14]. Aiming at a large sparse matrix A , the use of inverse iteration in Step 2 is proposed and analyzed in [16]. Nevertheless, the need for computing all the eigenvalues of $H(\epsilon)$ in Step 1 leaves Algorithm 2 inaccessible for larger matrices. Alternatively, an approach based on the convergence or divergence of the Chandrasekhar iteration [10], capable of exploiting sparsity, has been demonstrated to efficiently estimate the H_∞ norm of medium-sized systems.

III. FINDING EIGENVALUES ON THE IMAGINARY AXIS

In the following, we suppose that the matrix A is a large and sparse matrix, too large to admit a complete eigendecomposition. While Step 2 of Algorithm 2 can be effectively addressed by existing iterative eigenvalue solvers, see for example [2] and [26], the implementation of Step 1 requires more care. In this section, we address this issue by adapting a procedure developed in [15] for computing all the real eigenvalues of a general real matrix. It is based on exact inverse iterations [2], which benefit from the fact that the LU

decomposition of $H(\epsilon)$ can be computed efficiently by sparse direct factorization techniques, see, e.g., [12]. Moreover, we rely on the generically valid property of inverse iterations to converge to the eigenvalue nearest to the chosen shift.

To illustrate our approach, let us consider Figure 2, which displays the spectrum of a Hamiltonian matrix. Given an

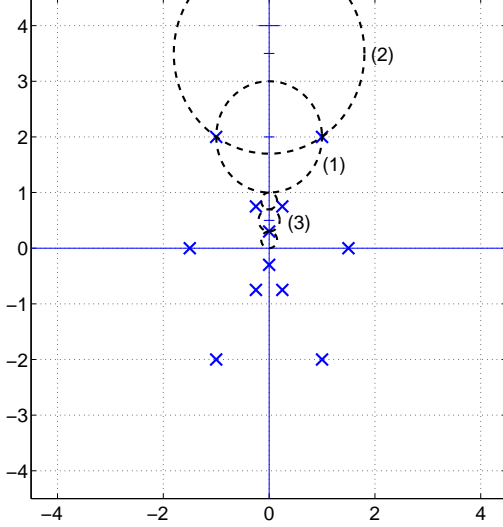


Fig. 2. Eigenvalues $\pm 1 \pm 2i, \pm 0.25 \pm 0.75i, \pm 0.3i, \pm 1.5i$ of a Hamiltonian matrix.

upper bound $u = 4$ on the largest imaginary part of the eigenvalues of H , we first apply inverse iteration with the shift $2i$ to discover the eigenvalues $\pm 1 + 2i$. This implies that there are no other eigenvalues in the disc marked by (1). We change the shift to $3.5i$, continue the iterations and rediscover the eigenvalues $\pm 1 + 2i$. Now we already know that there are no other eigenvalues in the union of the discs (1) and (2). In particular, there are no eigenvalues on the imaginary axis between i and $4i$. To investigate the interval between 0 and i we chose the shift $0.5i$ and detect the purely imaginary eigenvalue $0.3i$. The remaining parts are covered by two further iterations with shifts $0.85i$ and $0.15i$. Note that, due to the symmetry of the spectrum, there is no need to inspect the lower part of the imaginary axis.

For the general case, the algorithm is given by the following recursive procedure which is initially called with $l = 0$ and $u = \|H(\epsilon)\| + \epsilon$.

Algorithm 3 (Eigenvalue search on the imaginary axis):

Input: Hamiltonian matrix $H(\epsilon)$, lower bound l , upper bound u , with $l \geq 0$.

Output: Set Σ which contains all purely imaginary eigenvalues of $H(\epsilon)$ in the interval $[li, ui]$.

if $u < l$ **then** set $\Sigma = \emptyset$ and exit **end if**
Set $\sigma = (l + u)/2$.

1 Find eigenvalue λ of $H(\epsilon)$ that is closest to σi .

if $u - l < 2|\lambda - \sigma i|$ **then**
Set $\Sigma = \emptyset$ and exit.

else
% Inspect remaining upper/lower parts of imag. axis.

Call algorithm recursively to compute set Σ_u of eigenvalues in the interval $[(\sigma + |\lambda - \sigma i|)i, ui]$.

Call algorithm recursively to compute set Σ_l of eigenvalues in the interval $[li, (\sigma - |\lambda - \sigma i|)i]$.

Set $\Sigma = \Sigma_u \cup \Sigma_l$.

2 **if** $\text{real}(\lambda) = 0$ **then** $\Sigma = \Sigma \cup \{\lambda\}$ **end if**
end if

Note that each call of Step 1 requires a sparse LU factorization of $H(\epsilon) \pm \sigma i I$ for a different value of σ . Thus, to obtain reasonable performance, the number of recursions should be kept to a minimum. This issue will be addressed in the next section. In Step 2, the decision whether λ has zero real part must be met reliably despite the presence of approximation and roundoff error. For this purpose, it is important to use iterative methods that take the Hamiltonian structure of $H(\epsilon)$ into account.

A. Structure-preserving inverse iteration and Arnoldi methods

Given a shift σ with $\sigma i \notin \Lambda(H(\epsilon))$, the matrix

$$W = (H(\epsilon) - i\sigma I)^{-1}(H(\epsilon) + i\sigma I)^{-1} = (H^2 + \sigma^2 I)^{-1}$$

can be shown to be skew-Hamiltonian, i.e., it has the block structure

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{11}^T \end{bmatrix}, \quad W_{21}^T = -W_{21}, \quad W_{12}^T = -W_{12},$$

with $W_{11}, W_{21}, W_{12} \in \mathbb{R}^{n \times n}$. Moreover, the purely eigenvalues of H correspond to the real eigenvalues of W that are in the intervals $(-\infty, 0)$ or $(1/\sigma^2, \infty)$. Technical difficulties are caused by the fact that all eigenvalues of W are generically of algebraic multiplicity two. This may not only adversely affect the convergence of unstructured Arnoldi methods but it also implies that approximations to the real eigenvalues of W are generally not real. The reason is that arbitrarily small, real perturbations of W bifurcate its double real eigenvalues into complex conjugate pairs of eigenvalues. However, it can be shown that this effect cannot happen for double eigenvalues if the perturbations are restricted to be skew-Hamiltonian [23].

The skew-Hamiltonian implicitly restarted Arnoldi (SHIRA) method described in [29] produces eigenvalue approximations that are the exact eigenvalues of a perturbed skew-Hamiltonian matrix $W + E$. Therefore, double real eigenvalues of W are approximated along the real axis and, in turn, simple purely imaginary eigenvalues of H are approximated along the imaginary axis. Symplectic Lanczos methods [3], [35] share similar properties. One potential drawback of these Krylov subspace methods is that they are not guaranteed to converge first to the eigenvalue closest to the shift. In our numerical experiments, this effect was rarely observed; the sometimes slow overall convergence was found to be a larger concern.

IV. FINDING A GOOD UPPER BOUND

The performance of Algorithm 3 crucially depends on how often Step 1 is performed, i.e., how often the sparse LU decomposition of $H(\epsilon) \pm i\sigma I$ needs to be computed. This is mainly determined by the number of eigenvalues of $H(\epsilon)$ on or close to the imaginary axis. If ϵ is much larger than the distance to instability $\mu(A)$ then the pseudospectrum $\Lambda_\epsilon(A)$ is likely to cross the imaginary axis at several occasions. To illustrate this effect we generated a 200×200 pseudo-random matrix with the MATLAB command `rand(200)-4*eye(200)`, after having set the seed of the random generator to zero. The largest real part of the eigenvalues of A is -5.26×10^{-2} while $\mu(A) = 1.34 \times 10^{-2}$. A cheap upper bound for $\mu(A)$ is given by $\|A + A^T\|_F \approx 251$, where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. The following table contains the number n_H of purely imaginary eigenvalues of $H(\epsilon)$ for several values of ϵ .

ϵ	250	10	5	1	0.5	0.1	0.05
n_H	400	392	198	28	16	4	2

It can be seen that the cheap bound leads to an unacceptably high number of purely imaginary eigenvalues. Only an upper bound that is at most two orders larger than $\mu(A)$ results in a decent number. Using the largest real part of the eigenvalues yields the optimal value $n_H = 2$.

These observations suggest the following strategy for computing a tight upper bound.¹ Using ARPACK [26] we determine an orthogonal basis U for the invariant subspace of A belonging to a small number k of eigenvalues closest to the imaginary axis. Then the stability radius of the $k \times k$ compression $U^T A U$ can be cheaply determined, e.g., by Algorithm 2. Moreover, if E_{11} is a perturbation so that $U^T A U + E_{11}$ is not stable then $E = U E_{11} U^T$ is a perturbation so that $A + E$ is not stable. Since $\|E\| = \|E_{11}\|$, we thus have

$$\mu(A) \leq \mu(U^T A U).$$

In many cases, we observed that this bound is very tight already for rather small k , say $k = 4$. In our experiments, we used $k = 6$. Note that missing an eigenvalue closest to the imaginary axis does not affect the reliability of the overall stability radius computation, it only affects the efficiency of Algorithm 3. This allows the use of potentially less reliable methods for finding these eigenvalues, such as variants of the inexact Arnoldi and Jacobi-Davidson methods [25], [20].

V. NUMERICAL EXPERIMENTS

The aim of this section is to demonstrate that the presented approach can address the stability radius computation of larger matrices. Further testing and algorithmic improvements are needed to increase the robustness of the algorithm. Nevertheless, it is worth mentioning that each of the following computations took less than one minute using a preliminary MATLAB implementation on a 1.2GHz Intel x86 with 512 MByte memory. For solving the arising

¹One of the referees pointed out that a related approach has been discussed in [17].

sparse linear systems we used the UMFPACK [11] routines included in MATLAB. Examples A–D are taken from the Matrix Market collection [1] while E is a benchmark example used in LYAPACK [30]. Each section has a table containing the dimension of the matrix under consideration, the largest real part of all eigenvalues as detected by ARPACK, the total number of sparse direct factorizations needed in Algorithm 3, and the stability radius computed via our approach.

A. RDB1250

The matrices RDB1250 and RDB2048 represent central difference discretizations of a reaction-diffusion Brusselator model.

n	1250
max eigenvalue real part	-0.248
#factorizations	3
$\mu(A)$	8.46×10^{-2}

B. RDB2048

n	2048
max eigenvalue real part:	-0.247
#factorizations	3
$\mu(A)$	8.44×10^{-2}

C. BWM2000

BWM2000 arises from a finite difference discretization of the Brusselator wave model, shifted by $-3 \times 10^{-7} I$ to yield a stable matrix.

n	2000
max eigenvalue real part:	-5.57×10^{-8}
#factorizations	3
$\mu(A)$	2.52×10^{-8}

D. PDE2961

PDE2961 arises from a five-point central finite difference discretization of a two-dimensional linear elliptic equation, shifted by $-10I$ to yield a stable matrix.

n	2961
max eigenvalue real part:	-0.0929
#factorizations	13
$\mu(A)$	2.27×10^{-2}

E. FDM10000

FDM10000 represents a finite difference discretization of the 2D convection-diffusion heat equation

$$\Delta(u) - 10x \frac{\partial u}{\partial x} - 100y \frac{\partial u}{\partial y}$$

on the unit square.

n	10000
max eigenvalue real part:	-111.3
#factorizations	4
$\mu(A)$	35.5

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

An algorithm for computing the stability radius of a large and preferably sparse matrix has been proposed. Although it should be emphasized that several algorithmic details merit further investigation to increase its robustness, the new method has been demonstrated to address problems that were formerly intractable due to the sheer size of the involved matrices.

B. Future Works

A major focus of ongoing work is to develop less expensive and more robust algorithms for detecting purely imaginary eigenvalues of a Hamiltonian matrix. Structure-preserving Jacobi-Davidson methods [21] with additional safeguard strategies may offer a viable option. Another possibility for decreasing computational cost is the development of upper bounds that are cheaper to compute than the eigenvalues closest to the imaginary axis but still result in Hamiltonian matrices with few eigenvalues on or close to the imaginary axis. Finally we note that although extending the underlying ideas to H_∞ norm computation is straightforward, addressing the technical issues arising from the more complex structure of the associated Hamiltonian matrices is certainly a less trivial task.

Note that the definition of $\mu(A)$ in (1) admits complex perturbations even if the matrix A does not have complex entries. As pointed out by one of the referees, computing the more appropriate real stability radius [31] of a large real matrix is another open problem.

REFERENCES

- [1] Z. Bai, D. Day, J. W. Demmel, and J. J. Dongarra. A test matrix collection for non-Hermitian eigenvalue problems (release 1.0). Technical Report CS-97-355, Department of Computer Science, University of Tennessee, Knoxville, TN, USA, March 1997. Also available online from <http://math.nist.gov/MatrixMarket>.
- [2] Z. Bai, J. W. Demmel, J. J. Dongarra, A. Ruhe, and H. van der Vorst, editors. *Templates for the Solution of Algebraic Eigenvalue Problems*. Software, Environments, and Tools. SIAM, Philadelphia, PA, 2000.
- [3] P. Benner and H. Faßbender. An implicitly restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem. *Linear Algebra Appl.*, 263:75–111, 1997.
- [4] P. Benner and D. Kressner. Fortran 77 subroutines for computing the eigenvalues of Hamiltonian matrices II. To appear in *ACM Trans. Math. Software*. See also <http://www.tu-chemnitz.de/mathematik/hapack/>, 2004.
- [5] P. Benner, V. Mehrmann, and H. Xu. A numerically stable, structure preserving method for computing the eigenvalues of real Hamiltonian or symplectic pencils. *Numer. Math.*, 78(3):329–358, 1998.
- [6] S. Boyd and V. Balakrishnan. A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its L_∞ -norm. *Systems Control Lett.*, 15(1):1–7, 1990.
- [7] N. A. Bruinsma and M. Steinbuch. A fast algorithm to compute the H_∞ -norm of a transfer function matrix. *Sys. Control Lett.*, 14(4):287–293, 1990.
- [8] J. V. Burke, A. S. Lewis, and M. L. Overton. Robust stability and a criss-cross algorithm for pseudospectra. *IMA J. Numer. Anal.*, 23(3):359–375, 2003.
- [9] R. Byers. A bisection method for measuring the distance of a stable to unstable matrices. *SIAM J. Sci. Statist. Comput.*, 9:875–881, 1988.
- [10] Y. Chahlaoui, K. A. Gallivan, and P. Van Dooren. The h_∞ norm calculation for large sparse system. In *Proc. of Sixteenth International Symposium on Mathematical Theory of Networks and Systems (MTNS 2004)*, Leuven, Belgium, 2004.
- [11] T. A. Davis. Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Software*, 30(2):196–199, 2004.
- [12] T. A. Davis. *Direct Methods for Sparse Linear Systems*. SIAM, Philadelphia, PA, 2006.
- [13] J. W. Demmel. A counterexample for two conjectures about stability. *IEEE Trans. Automat. Control*, 32:340–342, 1987.
- [14] Y. Genin, P. Van Dooren, and V. Vermaut. Convergence of the calculation of H_∞ norms and related questions. In A. Beghi, L. Finesso, and G. Picci, editors, *Proceedings of the Conference on the Mathematical Theory of Networks and Systems, MTNS '98*, pages 429–432, 1998.
- [15] M. Gu, E. Mengi, M. L. Overton, J. Xia, and J. Zhu. Fast methods for estimating the distance to uncontrollability, 2005. To appear in *SIAM J. Matrix Anal. Appl.*
- [16] C. He and G. A. Watson. An algorithm for computing the distance to instability. *SIAM J. Matrix Anal. Appl.*, 20(1):101–116, 1999.
- [17] D. Hinrichsen, B. Kelb, and A. Linnemann. An algorithm for the computation of the structured complex stability radius. *Automatica J. IFAC*, 25(5):771–775, 1989.
- [18] D. Hinrichsen and A. J. Pritchard. Stability radii of linear systems. *Systems Control Lett.*, 7(1):1–10, 1986.
- [19] D. Hinrichsen and A. J. Pritchard. *Mathematical Systems Theory I*, volume 48 of *Texts in Applied Mathematics*. Springer-Verlag, 2005.
- [20] M. E. Hochstenbach. Variations on harmonic Rayleigh-Ritz for standard and generalized eigenproblems. Preprint, Dept. Math., Case Western Reserve University, September 2005.
- [21] M. E. Hochstenbach and D. Kressner. Jacobi-Davidson techniques for Hamiltonian eigenvalue problems, 2006. In preparation.
- [22] D. Kressner. *Numerical Methods and Software for General and Structured Eigenvalue Problems*. PhD thesis, TU Berlin, Institut für Mathematik, Berlin, Germany, 2004.
- [23] D. Kressner. Perturbation bounds for isotropic invariant subspaces of skew-Hamiltonian matrices. *SIAM J. Matrix Anal. Appl.*, 26(4):947–961, 2005.
- [24] D. Kressner and E. Mengi. Structure-exploiting methods for computing numerical and pseudospectral radii, 2006. In preparation.
- [25] R. B. Lehoucq and K. Meerbergen. Using generalized Cayley transformations within an inexact rational Krylov sequence method. *SIAM J. Matrix Anal. Appl.*, 20(1):131–148, 1999.
- [26] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK users' guide*. SIAM, Philadelphia, PA, 1998. Solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods.
- [27] K. Meerbergen and D. Roose. Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems. *IMA J. Numer. Anal.*, 16(3):297–346, 1996.
- [28] K. Meerbergen, A. Spence, and D. Roose. Shift-invert and Cayley transforms for detection of rightmost eigenvalues of nonsymmetric matrices. *BIT*, 34(3):409–423, 1994.
- [29] V. Mehrmann and D. S. Watkins. Structure-preserving methods for computing eigenpairs of large sparse skew-Hamiltonian/Hamiltonian pencils. *SIAM J. Sci. Comput.*, 22(6):1905–1925, 2000.
- [30] T. Penzl. *Lyapack users guide*. Technical report SFB393/00-33, Sonderforschungsbereich 393 Numerische Simulation auf massiv parallelen Rechnern, TU Chemnitz, 09107 Chemnitz, FRG, 2000.
- [31] L. Qiu, B. Bernhardsson, A. Rantzer, E. J. Davison, P. M. Young, and J. C. Doyle. A formula for computation of the real stability radius. *Automatica J. IFAC*, 31(6):879–890, 1995.
- [32] L. N. Trefethen and M. Embree. *Spectra and Pseudospectra. The Behavior of Nonnormal Matrices and Operators*. Princeton University Press, Princeton, NJ, 2005.
- [33] C. F. Van Loan. How near is a matrix to an unstable matrix? *Lin. Alg. and its Role in Systems Theory*, 47:465–479, 1984.
- [34] C. F. Van Loan. A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix. *Linear Algebra Appl.*, 61:233–251, 1984.
- [35] D. S. Watkins. On Hamiltonian and symplectic Lanczos processes. *Linear Algebra Appl.*, 385:23–45, 2004.