Transfer Functions and Path Following for Computing Pseudospectra

C. Bekas[*], *E. Gallopoulos*[†] *and V. Simoncini*[‡]

1 Introduction

The ϵ-pseudospectrum

$$\Lambda_\epsilon(A) = \{z \in \mathbb{C} : z \in \Lambda(A+E), \quad \|E\| \leq \epsilon\}, \tag{1}$$

of a non-normal matrix, $A \in \mathbb{C}^{n\times n}$, frequently furnishes useful information regarding its behavior in several applications [13]. Fig. 1 illustrates the eigenvalues and pseudospectrum boundaries $\partial\Lambda_\epsilon(A)$ corresponding to values $\epsilon = 10^{-1}, ..., 10^{-7}$ for the pentadiagonal Toeplitz matrix matrix `grcar(50)` from the Matrix Computation Toolbox for `MATLAB` [7]. An equivalent definition of $\Lambda_\epsilon(A)$ is based on the resolvent of A, $R(z) = (A - zI)^{-1}$:

$$\Lambda_\epsilon(A) = \{z \in \mathbb{C} : \|R(z)\| \geq \epsilon^{-1}\}. \tag{2}$$

Let now the matrix $A - zI$ have the singular value decomposition $A - zI = U\Sigma V^*$, where the symbol '*' denotes conjugate transposition. When the metric in use is the Euclidean norm $\|.\|_2$, an equivalent, third definition, is:

$$\Lambda_\epsilon(A) = \{z \in \mathbb{C} : \sigma_{\min}(A - zI) \leq \epsilon\}, \tag{3}$$

since $\|R(z)\| = \sigma_{\min}(A-zI)$, where $\sigma_{\min}(\cdot)$ denotes the smallest singular value of its matrix argument[1]. To obtain the pseudospectrum, we need practical methods for

[*]Computer Engineering & Informatics Department, University of Patras, Greece, e-mail:`knb@hpclab.ceid.upatras.gr`

[†]Computer Engineering & Informatics Department, University of Patras, Greece, e-mail `stratis@hpclab.ceid.upatras.gr`

[‡]Department of Mathematics, University of Bologna, Italy, `val@dragon.imati.cnr.it`

[1]For the remainder of this paper, $\|\cdot\|$ would refer to the Euclidean norm.

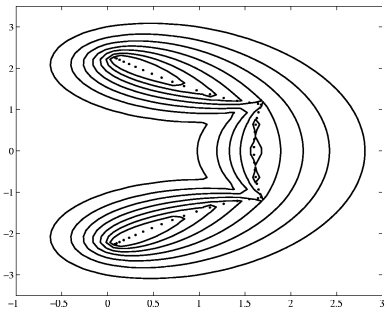

Figure 1. *Pseudospectra boundary curves* $\partial\Lambda_\epsilon(A)$, $\epsilon = 10^{-1}...10^{-7}$ *(solid lines) and eigenvalues (dots) of matrix* `grcar(50)`.

its computation and visualization. Relation (3) is behind `GRID`, a straightforward and robust but expensive algorithm for computing $\Lambda_\epsilon(A)$ [13]. This constructs a grid Ω_h over a region of the complex plane that includes $\Lambda_\epsilon(A)$, then computes $\sigma_{\min}(A - z_h I)$ for every node z_h of Ω_h. The pseudospectrum is built from a graphics postprocessing step. Unfortunately, it is well-known that as the size of the matrix and/or the number of gridpoints increases the cost of `GRID` becomes overwhelming: see [14, 4] for some efforts as well as the comprehensive repository [9].

2 The Transfer Function Framework

Let $A \in \mathbb{C}^{n\times n}$ and D^*, E be full rank matrices, typically rectangular, with row dimension n. Consider the projection of the resolvent $R(z)$ onto the subspaces spanned by the columns of matrices D^* and E, that is the transfer function

$$G_z(A, E, D^*) := D^* R(z) E.$$

Let $W_m = [w_1, \ldots, w_m]$ be the orthonormal basis for the Krylov subspace $\mathcal{K}_m(A, w_1)$, constructed by the Arnoldi iteration, according to the expanded relation:

$$AW_m = W_m H_{m,m} + h_{m+1,m} w_{m+1} e_m^*, \tag{4}$$

where $H_{m,m}$ is the square upper Hessenberg matrix consisting of the first m rows of $H_{m+1,m}$. We propose to use the approximation:

$$\|R(z)\| \approx \|G_z(A, W_{m+1}, W_{m+1}^*)\| = W_{m+1}^*(A - zI)^{-1} W_{m+1}. \tag{5}$$

The theoretical aspects of this approach were established in [11] while some initial practical experiences with the framework were presented at the 2000 Copper Mountain Conference on Iterative Methods; in [3] we illustrated some features of this framework in the context of a `MATLAB` based parallel environment. In this work we outline some new algorithms and results for the fast approximation of pseudospectra based on this framework in combination with existing domain-based

methods, including path-following. A direct calculation of $G_{z,m}(A)$ from formula (5) would require the solution of $m+1$ linear systems with $A - zI$, for each shift z. Nevertheless, as observed in [11] that if we define the vector

$$\phi_z = W_{m+1}^*(A - zI)^{-1}w_{m+1}, \tag{6}$$

where w_{m+1} is the last vector computed by the Arnoldi iteration, then

$$G_{z,m}(A) = [W_{m+1}^*(A - zI)^{-1}W_m, \phi_z] \in \mathbb{C}^{(m+1)\times(m+1)}.$$

Consider the computation of the first m columns of $G_{z,m}(A)$. From the Arnoldi factorization, it follows that

$$(A - zI)W_m = W_m(H_{m,m} - zI) + h_{m+1,m}w_{m+1}e_m^*. \tag{7}$$

Assuming that z is not an eigenvalue or Ritz value of A so as to make A or $H_{m,m}$ singular, pre-multiplying by $W_{m+1}^*(A - zI)^{-1}$ leads to

$$G_{z,m}(A) = [(\tilde{I} - h_{m+1,m}\phi_z e_m^*)(H_{m,m} - zI)^{-1}, \phi_z)]. \tag{8}$$

Consider next the application of the transfer function framework on `GRID`. That would require solving a single linear system $(A - z_hI)^{-1}v_{m+1}$ for each mesh point z_h. Krylov subspace linear solvers would be particularly suitable in this case because of the shift invariance of Krylov subspaces, i.e. $\mathcal{K}_m(A, b) = \mathcal{K}_m(A - zI, b)$, for every starting vector b and shift $z \in \mathbb{C}$. Since pseudospectra are interesting only for non-normal matrices, it is natural to consider using `GMRES` [10] and its handling of shift-invariance, e.g. [6]. Table 1 depicts the resulting transfer functions - grid method, which we call `TRGRID`. In Figure 2 we compare `TRGRID` with the method presented in [12] which is based on `GRID` and the approximation of $\sigma_{\min}(A - z_I)$ by $\sigma_{\min}(\tilde{H}_m - z_I)$, where $\tilde{H}_m$ is the augmented Hessenberg matrix resulting from the Arnoldi iteration. We call this method `AHGRID`. The test matrix was `gre_1107` for the Matrix Market and $\epsilon = 0.1, 0.01$. Solid lines are the results of `GRID` using the `svds` routine of `MATLAB`, which is based on `ARPACK` and uses shift and invert Lanczos to compute the smallest singular value. Table 2 depicts runtimes on a `PIII @ 866 MHz` workstation with `1Gb RAM`, running `Windows 2000`.

3 Path Following and Transfer Functions

One important class of domain-based methods uses numerical path following to trace curves that define the pseudospectrum, in particular, boundaries $\partial\Lambda_\epsilon(A)$ for any given ϵ. The first such algorithm was presented by Brühl in [5], who demonstrated impressive savings of path following compared to `GRID`. Further work, in [1], advanced the original path following approach, into an algorithm called `Cobra`; this permitted the effective use of path following on parallel systems while achieving greater robustness. Two other algorithms that lend themselves to parallel implementation are `PAT` [8], that is based on triangulation and bisection; and `PsDM` [2] that applies path following in directions transversal to the pseudospectrum boundaries.

TRGRID(m, d) **algorithm**
(Input *)*
Points $z_i, i = 1, ..., M$, vector w_1
with $\|w_1\| = 1$, scalars m, d.
1. $[W_{m+1}, H_{m+1,m}] \leftarrow$ `arnoldi` (A, w_1, m)
2. $[\hat{W}_{d+1}, F_{d+1,d}] \leftarrow$ `arnoldi`(A, w_{m+1}, d)
3. **for** $i = 1, ..., M$
4. $\quad Y(:, i) = \text{argmin}_y\{\|(F_{d+1,d} - z_i\tilde{I}_d)y - e_1\|\}$
5. **end**
6. Compute $\Phi_z = W^*_{m+1}\hat{W}_d Y$
7. **for** $i = 1, ..., M$
8. $\quad D_i = (\tilde{I} - h_{m+1,m}\Phi_z(:, i)e^*_m)(H_{m,m} - z_i I)^{-1}$
9. $\quad \|G_{z_i}(A)\| = \|[D_i, \phi_{z_i}]\|$
10. **end**

Table 1. TRGRID *for* M *points* z *using* GMRES *dimension* d *and Transfer Function dimension* m.

Method	Runtimes (secs)
GRID (svds)	3438
AHGRID(100)	49.5
AHGRID(150)	221
AHGRID(200)	574
TRGRID$(100, 50)$	111

Table 2. *Runtimes for* TRGRID *and* AHGRID *for matrix* gre_1107 *on a* 25×50 *mesh for the domain* $\Omega = [-1, 1.5] \times [0, 1]$.

Except for PAT, the critical component of the aforementioned algorithms is Newton iteration, applied to solve the nonlinear equation[2]

$$\mathcal{F}(z) - \epsilon = 0, \quad \text{where} \quad \mathcal{F}(z) = \sigma_{\min}(A - zI) \tag{9}$$

for any $\epsilon > 0$. Therefore, it becomes imperative to compute $\nabla\mathcal{F}(x, y)$.

Assume that it is possible to construct the full length Krylov subspace $\mathcal{K}_n(A, w_1)$, and that W_n is the corresponding orthonormal basis, where n is the size of matrix A and w_1 is a starting vector of unit norm. Define Q as follows:

$$W_n = [W_{m+1}, Q]$$

[2]We would be identifying the complex plane $\mathbb{C}$ with $\mathbb{R}^2$ and frequently use $\mathcal{F}(z)$ to denote $\mathcal{F}(x, y) : \mathbb{R}^2 \to \mathbb{R}$ that is real analytic at simple nonzero singular values.

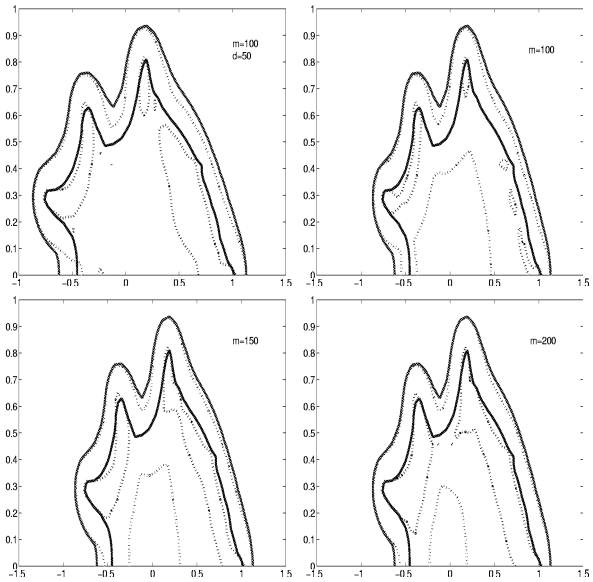

Figure 2. *Experiments with* `TRGRID` *and* `AHGRID` *for matrix* `gre_1107` *and* $\epsilon = 0.1, 0.01$. *Left-top corner:* `TRGRID`$(100, 50)$. *Rest:* `AHGRID`(100), `AHGRID`(150), `AHGRID`(200).

Lemma 1. *Using the same notation as above, the following relation holds:*

$$\overline{\nabla \|G_{z,m}(A)\|} = \sigma_1^2 v_1^* u_1 + u_1^* W_{m+1}^* (A - zI)^{-1} Q Q^* (A - zI)^{-1} W_{m+1} v_1,$$

where (σ_1, u_1, v_1) *is the largest singular triplet of* $G_{z,m}(A)$.

The following lemma immediately suggests an approximation for the gradient of the transfer function:

Lemma 2. *Let* (u_1, σ_1, v_1) *the maximum singular value triplet of* $G_{z,m}(A)$. *Let* $r_r = (A - zI)^{-1} W_{m+1} v_1 - \sigma_1 W_{m+1} u_1$ *and* $r_l = (A - zI)^{-*} W_{m+1} u_1 - \sigma_1 W_{m+1} v_1$ *be the right and left residuals, respectively. Then*

$$|\overline{\nabla \|G_{z,m}(A)\|} - \sigma_1 v_1^* u_1| \leq \|r_r\| \, \|r_l\|.$$

We next investigate the effectiveness of the approximation. For `grcar`(100), we consider points that lie on $\partial\Lambda_\epsilon(A)$ for $\epsilon = 10^{-k}$, where k takes values $1, 6, 15$ as indicated in parentheses in the caption. Similarly, the points selected for `kahan` were on pseudospectrum curves corresponding to $k = -1, -2, -2.5$. Table 3 shows the extent to which (I) achieves the desired goals. Approximation of the norm of the resolvent with transfer functions is a theoretically powerful approach for

matrix: grcar(100)						
	$\lvert\frac{\lVert(A-zI)^{-1}\rVert_2-\lVert G_{z,m}(A)\rVert_2}{\lVert(A-zI)^{-1}\rVert_2}\rvert$			$\lvert\frac{\nabla\lVert(A-zI)^{-1}\rVert_2-\nabla\lVert G_{z,m}(A)\rVert_2}{\nabla\lVert(A-zI)^{-1}\rVert_2}\rvert$		
m	z_1	z_2	z_3	z_1	z_2	z_3
30	1.2	2.5	4.1	8.9e-1	7e-1	8e-1
50	4.3e-1	7.7e-1	1.9	3.5e-1	4.6e-1	6.5e-1
70	3.1e-3	5.2e-1	5.1e-1	6.9e-3	3.6e-1	3.3e-1
90	8.9e-4	3.3e-1	9e-2	2e-3	2.6e-1	8.4e-2
	$\lvert\frac{\mathrm{I}-\nabla\lVert G_{z,m}(A)\rVert_2}{\nabla\lVert G_{z,m}(A)\rVert_2}\rvert$			$\lvert\frac{\mathrm{II}}{\mathrm{I}}\rvert$		
m	z_1	z_2	z_3	z_1	z_2	z_3
30	1.9e-1	709	1.4e11	1.7e-1	1	9.9e-1
50	3.1e-1	82	1.6e11	4.4e-1	9.9e-1	1
70	7.7e-6	4.7e-3	1.1e7	1.1e-5	1.4	29.2
90	1.8e-6	4.7e-5	7.4e2	1.9e-6	1.3	4.3e5

matrix: kahan(100)						
	$\lvert\frac{\lVert(A-zI)^{-1}\rVert_2-\lVert G_{z,m}(A)\rVert_2}{\lVert(A-zI)^{-1}\rVert_2}\rvert$			$\lvert\frac{\nabla\lVert(A-zI)^{-1}\rVert_2-\nabla\lVert G_{z,m}(A)\rVert_2}{\nabla\lVert(A-zI)^{-1}\rVert_2}\rvert$		
m	z_1	z_2	z_3	z_1	z_2	z_3
30	4.6e-2	6.5e-6	9.2e-2	5.1e-2	1.2e-5	9.2e-2
50	3.1e-2	3.9e-6	5.2e-2	3.5e-2	6.9e-6	5.3e-2
70	6.7e-3	6.5e-7	8.3e-3	7.6e-3	1.2e-6	8.8e-3
90	1.8e-4	2.3e-9	2.9e-5	2.1e-4	4e-9	3e-5
	$\lvert\frac{\mathrm{I}-\nabla\lVert G_{z,m}(A)\rVert_2}{\nabla\lVert G_{z,m}(A)\rVert_2}\rvert$			$\lvert\frac{\mathrm{II}}{\mathrm{I}}\rvert$		
m	z_1	z_2	z_3	z_1	z_2	z_3
30	3.8e-6	2.9e-9	2.3e-5	3.8e-6	2.6e-9	2.3e-5
50	1.3e-7	1.2e-9	4.8e-5	1.3e-7	1e-9	4.9e-5
70	6.7e-8	3.4e-10	4.3e-6	6.7e-8	9.4e-11	4.3e-6
90	1.1e-7	2.9e-10	1.6e-9	1.1e-7	1e-13	1.9e-9

Table 3. *Top: Approximation of gradient of transfer function norm for* `grcar(100)` *and points* $z_1 = -0.6034 + 1.6379i$ (-1), $z_2 = 1.8103 + 1.4655i$ (-6), $z_3 = 0.4310 + 1.8103i$ (-15). *Bottom: Approximation of gradient of transfer function norm for* `kahan(100)`. $z_1 = 0.0862 + 1.2931i$ (-1), $z_2 = 0.7759 - 0.2586i$ (-2), $z_3 = -0.6034 + 0.2586i$ (-2.5). *The numbers in parenthesis are equal to* $\log_{10}(\sigma_{\min}(A - z_iI), i = 1, 2, 3$.

approximating pseudospectra. Our experiments show that transfer functions can be efficiently combined with existing methods such as `GRID` and path following. In our current research we investigate the incorporation of restarted Krylov linear solvers in the transfer function framework that will allow us to tackle very large problems.

Bibliography

[1] C. Bekas and E. Gallopoulos. Cobra: Parallel path following for computing the matrix pseudospectrum. *Parallel Computing*, 27(14):1879–1896, 2001.

[2] C. Bekas and E. Gallopoulos. Parallel computation of pseudospectra by fast descent. *Parallel Computing*, 28:223–242, 2002.

[3] C. Bekas, E. Kokiopoulou, E. Gallopoulos, and V. Simoncini. Parallel computation of pseudospectra using transfer functions on a `MATLAB-MPI` cluster platform. In D. Kranzlmueller, P. Kacsuk, J. Dongarra, and J. Volkert, editors, *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, volume 2474 of *Lecture Notes in Computer Science*, pages 199–207. Springer-Verlag, 2002.

[4] C. Bekas, E. Kokiopoulou, I. Koutis, and E. Gallopoulos. Towards the effective parallel computation of matrix pseudospectra. In *Proc. 15th ACM Int'l. Conf. Supercomputing (ICS'01), Sorrento, Italy*, pages 260–269, June 2001.

[5] M. Brühl. A curve tracing algorithm for computing the pseudospectrum. *BIT*, 33(3):441–445, 1996.

[6] A. Frommer and U. Glässner. Restarted GMRES for shifted linear systems. *SIAM J. Sci. Comput.*, 19(1):15–26, January 1998.

[7] N.J. Higham. The matrix computation toolbox. Technical report, Manchester Centre for Computational Mathematics. In `www.ma.man.uc.uk/~higham/mctoolbox`.

[8] D. Mezher and B. Philippe. Parallel computation of the pseudospectrum of large matrices. *Parallel Computing*, 28(2):199–221, 2002.

[9] Pseudospectra gateway. At the Oxford University site http://web.comlab.ox.ac.uk/projects/pseudospectra.

[10] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, July 1986.

[11] V. Simoncini and E. Gallopoulos. Transfer functions and resolvent norm approximation of large matrices. *Electronic Transactions on Numerical Analysis (ETNA)*, 7:190–201, 1998.

[12] K.-C. Toh and L.N. Trefethen. Calculation of pseudospectra by the Arnoldi iteration. *SIAM J. Sci. Comput.*, 17(1):1–15, 1996.

[13] L.N. Trefethen. Computation of pseudospectra. In *Acta Numerica 1999*, volume 8, pages 247–295. Cambridge University Press, 1999.

[14] T. Wright and L. N. Trefethen. Large-scale computation of pseudospectra using ARPACK and Eigs. *SIAM J. Sci. Stat. Comput.*, 23(2):591:605, 2001.