

Article

Eigenvalue Estimates via Pseudospectra †

Georgios Katsouleas ^{*,‡}, Vasiliki Panagakou [‡] and Panayiotis Psarrakos [‡]

Department of Mathematics, Zografou Campus, National Technical University of Athens, 15773 Athens, Greece; vpanagakou@mail.ntua.gr (V.P.); ppsarr@math.ntua.gr (P.P.)

* Correspondence: g_katsouleas@mail.ntua.gr

† This paper is dedicated to Mr. Constantin M. Petridi.

‡ These authors contributed equally to this work.

Abstract: In this note, given a matrix $A \in \mathbb{C}^{n \times n}$ (or a general matrix polynomial $P(z)$, $z \in \mathbb{C}$) and an arbitrary scalar $\lambda_0 \in \mathbb{C}$, we show how to define a sequence $\{\mu_k\}_{k \in \mathbb{N}}$ which converges to some element of its spectrum. The scalar λ_0 serves as initial term ($\mu_0 = \lambda_0$), while additional terms are constructed through a recursive procedure, exploiting the fact that each term μ_k of this sequence is in fact a point lying on the boundary curve of some pseudospectral set of A (or $P(z)$). Then, the next term in the sequence is detected in the direction which is normal to this curve at the point μ_k . Repeating the construction for additional initial points, it is possible to approximate peripheral eigenvalues, localize the spectrum and even obtain spectral enclosures. Hence, as a by-product of our method, a computationally cheap procedure for approximate pseudospectra computations emerges. An advantage of the proposed approach is that it does not make any assumptions on the location of the spectrum. The fact that all computations are performed on some dynamically chosen locations on the complex plane which converge to the eigenvalues, rather than on a large number of predefined points on a rigid grid, can be used to accelerate conventional grid algorithms. Parallel implementation of the method or use in conjunction with randomization techniques can lead to further computational savings when applied to large-scale matrices.

Keywords: pseudospectra; eigenvalues; matrix polynomial; perturbation; Perron root; large-scale matrices; approximation algorithm



Citation: Katsouleas, G.; Panagakou, V.; Psarrakos, P. Eigenvalue Estimates via Pseudospectra. *Mathematics* **2021**, *9*, 1729. <https://doi.org/10.3390/math9151729>

Academic Editors: Khalide Jbilou and Marilena Mitrouli

Received: 30 May 2021
Accepted: 18 July 2021
Published: 22 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The theory of pseudospectra originates in numerical analysis and can be traced back to Landau [1], Varah [2], Wilkinson [3], Demmel [4], and Trefethen [5], motivated by the need to obtain insights into systems evolving in ways that the eigenvalues alone could not explain. This is especially true in problems where the underlying matrices or linear operators are non-normal or exhibit in some sense large deviations from normality. A better understanding of such systems can be gained through the concept of pseudospectrum, which, for a matrix $A \in \mathbb{C}^{n \times n}$ and a positive parameter $\epsilon > 0$, was introduced as the subset of the complex plane that is bounded by the ϵ^{-1} -level set of the norm of the resolvent $\|(\mu I - A)^{-1}\|$. A second definition stated in terms of perturbations characterizes the elements of this set as eigenvalues of some perturbation $A + E$ with $\|E\| \leq \epsilon$. In this sense, the notion of pseudospectrum provides information that goes beyond eigenvalues, while retaining the advantage of being a natural extension of the spectral set. In fact, for different values of magnitude ϵ , pseudospectrum provides a global perspective on the effects of perturbations; this is in stark contrast to the concept of condition number, where only the worst-case scenario is considered.

On one hand, pseudospectrum may be used as a visualization tool to reveal information regarding the matrix itself and the sensitivity of its eigenvalues. Applications within numerical analysis include convergence of nonsymmetric matrix iterations [6], backward error analysis of eigenvalue algorithms [7], and stability of spectral methods [8]. On the

other hand, it is a versatile tool that has been used to obtain quantitative bounds on the transient behavior of differential equations in finite time, which may deviate from the long-term asymptotic behavior [9]. Important results involving pseudospectra have been also been obtained in the context of spectral theory and spectral properties of banded Toeplitz matrices [10,11]. Although emphasis has been placed on the standard eigenproblem, attention has also been drawn to matrix pencils [12] and more general matrix polynomials [13,14] arising in vibrating systems, control theory, etc. For a comprehensive overview of this research field and its applications, the interested reader may refer to [15].

In this note, we propose an application of pseudospectral sets as a mean to obtain eigenvalue estimates in the vicinity of some complex scalar. In particular, given a matrix (or a general matrix polynomial) and a scalar $\lambda_0 \in \mathbb{C}$, we construct a sequence $\{\mu_k\}_{k \in \mathbb{N}}$ that converges to some element of its spectrum. The scalar λ_0 serves as initial term ($\mu_0 = \lambda_0$), while additional terms are constructed through an iterative procedure, exploiting the fact that each term μ_k of this sequence is in fact a point lying on the boundary curve of some pseudospectral set. Then, the next term in the sequence is detected in the perpendicular direction to the tangent line at the point μ_k . Repeating the construction for a tuple of initial points encircling the spectrum, several peripheral eigenvalues are approximated. Since the pseudospectrum may be disconnected, this procedure allows the identification of individual connected components and, as a by-product, a convenient and numerically efficient procedure for approximate pseudospectrum computation emerges. Moreover, this approach is clearly amenable to parallelization or randomization and can lead to significant computational savings when applied to problems involving large-scale matrices.

Our paper is organized as follows. In Section 2, we provide the necessary theoretical background on the method and provide examples for the constant matrix case. As confirmed by numerical experiments, the method can provide a sufficiently accurate pseudospectrum computation at a much-reduced computational cost, especially in cases where the spectrum is convexly independent (i.e., each eigenvalue does not lie in the convex hull of the others) or exhibits large eigenvalue gaps. A second application of the method on Perron-root approximation for non-negative matrices is presented. Then, Section 3 shows how the procedure may be modified to estimate the spectrum of more general matrix polynomials. Numerical experiments showcasing the application of the method on damped mass-spring and gyroscopic systems conclude the paper.

2. Eigenvalues via Pseudospectra

Let the matrix $A \in \mathbb{C}^{n \times n}$ with spectrum $\sigma(A) = \{\mu \in \mathbb{C} : \det(\mu I - A) = 0\}$, where $\det(\cdot)$ denotes the *determinant* of a matrix. With respect to the $\|\cdot\|_2$ -norm, the *pseudospectrum* of A is defined by

$$\begin{aligned}\sigma_\epsilon(A) &= \left\{ \mu \in \mathbb{C} : \frac{1}{\|(\mu I - A)^{-1}\|_2} \leq \epsilon \right\} \\ &= \left\{ \mu \in \mathbb{C} : \mu \in \sigma(A + E) \text{ for some } E \in \mathbb{C}^{n \times n} \text{ with } \|E\| \leq \epsilon \right\} \\ &= \left\{ \mu \in \mathbb{C} : s_{\min}(\mu I - A) \leq \epsilon \right\},\end{aligned}$$

where $s_{\min}(\cdot)$ denotes the smallest *singular value* of a matrix and $\epsilon > 0$ is the maximum norm of admissible perturbations.

For every choice of increasing positive parameters $0 < \epsilon_1 < \epsilon_2 < \epsilon_3 < \dots$, the corresponding closed, strictly nested sequence of pseudospectra

$$\sigma_{\epsilon_1}(A) \subset \sigma_{\epsilon_2}(A) \subset \sigma_{\epsilon_3}(A) \subset \dots$$

is obtained. In fact, the respective boundaries satisfy the inclusions

$$\begin{aligned} \partial\sigma_{\epsilon_1}(A) &\subseteq \{\mu \in \mathbb{C} : s_{\min}(\mu I - A) = \epsilon_1\} \\ \partial\sigma_{\epsilon_2}(A) &\subseteq \{\mu \in \mathbb{C} : s_{\min}(\mu I - A) = \epsilon_2\} \\ \partial\sigma_{\epsilon_3}(A) &\subseteq \{\mu \in \mathbb{C} : s_{\min}(\mu I - A) = \epsilon_3\} \\ &\vdots \qquad \qquad \qquad \vdots \end{aligned}$$

It is also clear that, for any $\lambda \in \sigma(A)$, $s_{\min}(\lambda I - A) = 0$.

Our objective now is to exploit the properties of these sets to detect an eigenvalue of A in the vicinity of a given scalar $\lambda_0 \in \mathbb{C} \setminus \sigma(A)$. This given point of interest may be considered to lie on the boundary of some pseudospectral set, i.e., there exists some non-negative parameter $\hat{\epsilon}_1 > 0$, such that

$$\lambda_0 \in \partial\sigma_{\hat{\epsilon}_1}(A) \subseteq \{\mu \in \mathbb{C} : s_{\min}(\mu I - A) = \hat{\epsilon}_1\}. \tag{1}$$

Indeed, points satisfying the equality $s_{\min}(\mu I - A) = \epsilon$ for some $\epsilon > 0$ and lying in the interior of $\sigma_{\epsilon}(A)$ are finite in number. Thus, in the generic case, we may think of the inclusion (1) as an equality.

We consider the real-valued function $g_A : \mathbb{C} \rightarrow \mathbb{R}_+$ with $g_A(z) = s_{\min}(zI - A)$. In the process of formulating a curve-tracing algorithm for pseudospectrum computation [16], Brühl analyzed $g_A(z)$ and, identifying $\mathbb{C} \equiv \mathbb{R}^2$, noted that its differentiability is explained by the following Theorem in [17]:

Theorem 1. *Let the matrix valued function $P(\chi) : \mathbb{R}^d \rightarrow \mathbb{C}^{n \times n}$ be real analytic in a neighborhood of $\chi_0 = (x_0^1, \dots, x_0^d)$ and let σ_0 a simple nonzero singular value of $P(\chi_0)$ with u_0, v_0 its associated left and right singular vectors, respectively.*

Then, there exists a neighborhood \mathcal{N} of χ_0 on which a simple nonzero singular value $\sigma(\chi)$ of $P(\chi)$ is defined with corresponding left and right singular vectors $u(\chi)$ and $v(\chi)$, respectively, such that $\sigma(\chi_0) = \sigma_0$, $u(\chi_0) = u_0$, $v(\chi_0) = v_0$ and the functions σ, u, v are real analytic on \mathcal{N} . The partial derivatives of $\sigma(\chi)$ are given by

$$\frac{\partial s(\chi_0)}{\partial \chi^j} = \operatorname{Re} \left(u_0^* \frac{\partial P(\chi_0)}{\partial \chi^j} v_0 \right), \quad j = 1, \dots, d.$$

Hence, recalling (1) and assuming $\hat{\epsilon}_1$ is a simple singular value of the matrix $P(\lambda_0) = \lambda_0 I - A$, then

$$\nabla s_{\min}(zI - A)|_{z=\lambda_0} = (\operatorname{Re}(v_{\min}^* u_{\min}), \operatorname{Im}(v_{\min}^* u_{\min})) = v_{\min}^* u_{\min},$$

where u_{\min} and v_{\min} denote the left and right singular vectors of $\lambda_0 I - A$ associated to $\hat{\epsilon}_1 = s_{\min}(\lambda_0 I - A)$, respectively [16] (Corollary 2.2).

On the other hand, if λ is an eigenvalue of A near λ_0 , it holds $|\lambda - \lambda_0| \leq \hat{\epsilon}_1$. The latter observation follows from the fact that

$$\begin{aligned} \sigma_{\epsilon}(A) &\supseteq \sigma(A) + D(0, \epsilon) \\ &= \{z \in \mathbb{C} : \operatorname{dist}(z, \sigma(A)) \leq \epsilon\}, \end{aligned}$$

where $D(0, \epsilon) = \{z \in \mathbb{C} : |z| \leq \epsilon\}$ and equality holds for normal matrices. So, the scalar

$$\begin{aligned} \mu_1 &= \lambda_0 - \hat{\epsilon}_1 \cdot \frac{\nabla s_{\min}(zI - A)}{|\nabla s_{\min}(zI - A)|} \Big|_{z=\lambda_0} \\ &= \lambda_0 - s_{\min}(\lambda_0 I - A) \cdot \frac{v_{\min}(\lambda_0)^* u_{\min}(\lambda_0)}{|v_{\min}(\lambda_0)^* u_{\min}(\lambda_0)|} \end{aligned}$$

the procedure and properly trace its boundary. Moreover, cases where pseudospectrum computation is required for a whole tuple of parameters ϵ drastically compromise the efficiency of path-following algorithms.

Our approach is to use the approximating sequences (2) to decrease the number of singular value evaluations and therefore speed up the computation of pseudospectra. The basic steps are outlined as follows:

- i. Select a tuple of initial points $\{\mu_0^j\}_{j=1}^s \in \mathbb{C}$ encircling the spectrum; for instance, these can be chosen on the circle $\{z \in \mathbb{C} : |z| = \|A\|\}$.
- ii. Construct eigenvalue approximating sequences $\{\mu_k^j\}_{k=0}^{n_j}$ ($j = 1, \dots, s$), as in (2). If $\epsilon_k^j > 0$ ($k = 1, \dots, n_j$) are such that $\mu_k^j \in \partial\sigma_{\epsilon_k^j}(A)$, the length n_j of each sequence is determined, so that $s_{\min}(\mu_{n_j}^j I - A) \leq \epsilon_0$ for all $j = 1, \dots, s$, where ϵ_0 is some prefixed parameter value. In other words, ϵ_0 indicates the tolerance with which the approached by the constructed sequences eigenvalues should be approximated and corresponds to the minimum parameter for which pseudospectra will be computed.
- iii. Classify the sequences into distinct clusters, according to the proximity of their final terms. This step may be performed using a k-means clustering algorithm, using a suitable criterion to evaluate the optimal number of groups.
- iv. Compute

$$u = \min_{j=1, \dots, s} \max_{j=1, \dots, n_j} \epsilon_k^j (> \epsilon_0) \text{ and } \ell = \max_{j=1, \dots, s} \min_{j=1, \dots, n_j} \epsilon_k^j (< \epsilon_0).$$

- v. If necessary, repeat the procedure for t additional points between the centroids of the detected clusters, constructing additional sequences, so that

$$\min_{j=s+1, \dots, s+t} \max_{j=1, \dots, n_j} \epsilon_k^j > u \text{ and } \max_{j=s+1, \dots, s+t} \min_{j=1, \dots, n_j} \epsilon_k^j < \ell.$$

- vi. Detect boundary points of $\sigma_\epsilon(A)$ for any choice of parameters $\epsilon \in [\ell, u]$ along the polygonal chains formed by the total of $s + t$ constructed sequences of points by interpolation.
- vii. Fit closed spline curves passing through the respective sets of boundary points in $\partial\sigma_\epsilon(A)$ for the various choices of $\epsilon \in [\ell, u]$ to obtain sketches of the corresponding pseudospectra $\sigma_\epsilon(A)$.

The proposed method successfully localizes the spectrum, initiating the procedure with a restricted number of points. Then, singular value computations are kept to a minimum by considering points only on the constructed sequences. Pseudospectrum components corresponding to *peripheral* eigenvalues $\lambda \notin \text{co}(\sigma(A) \setminus \{\lambda\})$ not in the convex hull of the other eigenvalues, are thus extremely easy to identify. This approach is also well-suited to cases, where the matrix has *convexly independent* spectrum; i.e., when $\lambda \notin \text{co}(\sigma(A) \setminus \{\lambda\})$, for every $\lambda \in \sigma(A)$. Moreover, it is clearly amenable to parallelization, which could lead to significant computational savings in cases of large matrices.

Application 1. We consider a random matrix $A \in \mathbb{C}^{6 \times 6}$, the sole constraint being that its eigenvalues are distant from each other; the real and imaginary parts of its entries follow the standardized normal distribution scaled by 10^4 . For the proposed procedure, we select initial points $\{\mu_0^j\}_{j=1}^{10} \subset \{z \in \mathbb{C} : |z| = \|A\|_1\}$ and exploit the fact that the corresponding sequences $\{\mu_k^j\}_{k=0}^{n_j}$ generated as in (2) converge to some element of $\sigma(A)$. The number n_j of terms in each sequence ($j = 1, \dots, 10$) is determined, so that all values $\{s_{\min}(\mu_{n_j}^j I - A)\}_{j=1}^{10}$ do not exceed $\epsilon_0 = 0.5$. The sequences are organized into distinct clusters, grouping together those sequences which approximate the same element of $\sigma(A)$. This grouping is performed using a k-means clustering algorithm, where

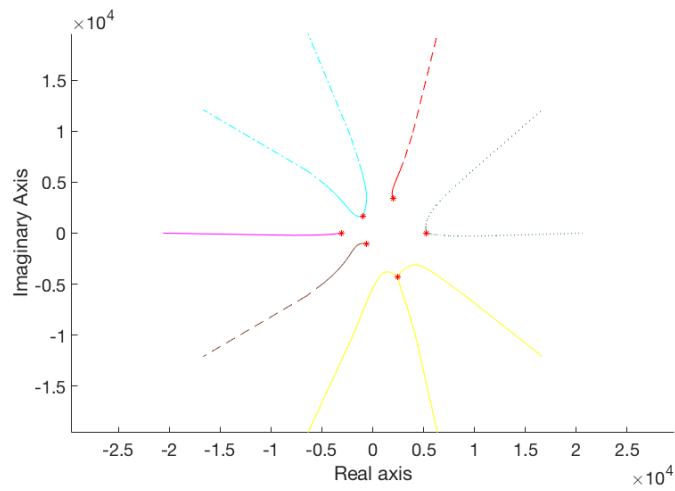
the optimal number of clusters is evaluated via the silhouette criterion and using a distance metric based on the sum of absolute differences between points. Since six different groups are identified, clearly all elements of $\sigma(A)$ have been sufficiently approximated by at least one of the sequences. For an illustration, refer to Figure 1a; different colors have been used to differentiate between polygonal chains corresponding to distinct clusters. The construction so far required 914 singular value computations. Having calculated all parameter values ϵ_k^j such that $\mu_k^j \in \partial\sigma_{\epsilon_k^j}(A)$ during the previous procedure, it is possible to interpolate between these known points along the trajectories formed by $\{\mu_k^j\}_{k=0}^{n_j}$ ($j = 1, \dots, 10$) to approximate boundary points of $\partial\sigma_\epsilon(A)$ for selected values $\epsilon > \epsilon_0 = 0.5$. Since all ten trajectories converge to eigenvalues from points encircling $\sigma(A)$, to obtain better pseudospectra approximations, it is necessary to repeat the procedure for additional suitably selected points. Hence, for each cluster we consider three additional points; see Figure 1b. In particular, denoting c_1, \dots, c_6 the centroids of the clusters, for each j we consider the three centroids $\{c_{j,k}\}_{k=1}^3$ which lie closest to c_j and take the convex combinations

$$p_k^j = \frac{1}{6}(5c_j + c_{j,k}), \quad k = 1, 2, 3.$$

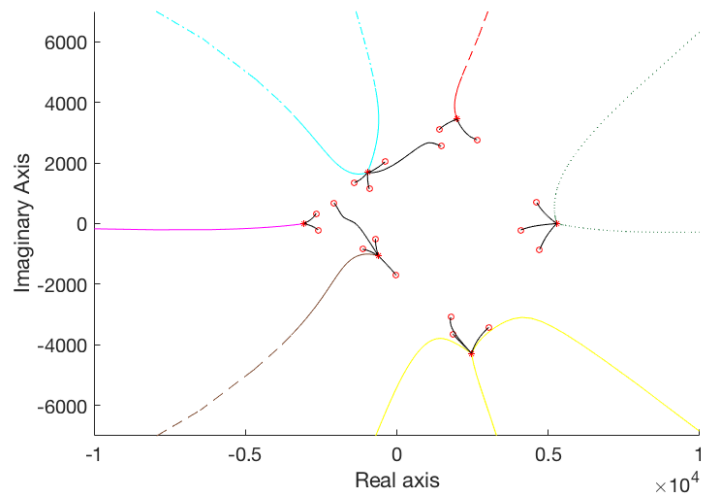
Then, additional sequences corresponding to these extra points are constructed so that the desired parameter values of ϵ for which pseudospectra should be computed (in this instance, the triple of $\epsilon = 1, 5, 10 \in [\ell, u]$) may be interpolated within these trajectories, as for the ten initial ones. This imposes an extra cost of 1170 additional singular value computations (2084 in total). The resulting approximations of the pseudospectra components identified by the upper left corner trajectories for $\epsilon = 1, 5, 10$ are depicted in greater detail in Figure 1c; the relevant eigenvalue is indicated by “*”.

An advantage of this procedure is that it does not require some a-priori knowledge of the initial region Ω on the complex plane where the spectrum is located. In fact, the very nature of this specific example, whose spectrum covers a wide area Ω , would render computations on a suitable grid impractical. Another way in which this method diverges from conventional grid algorithms is in that the computations are performed on a dynamically chosen set of points, iteratively selected as the corresponding trajectories converge to peripheral eigenvalues and identify the relevant pseudospectrum components, rather than on a large number of predefined points on a rigid grid.

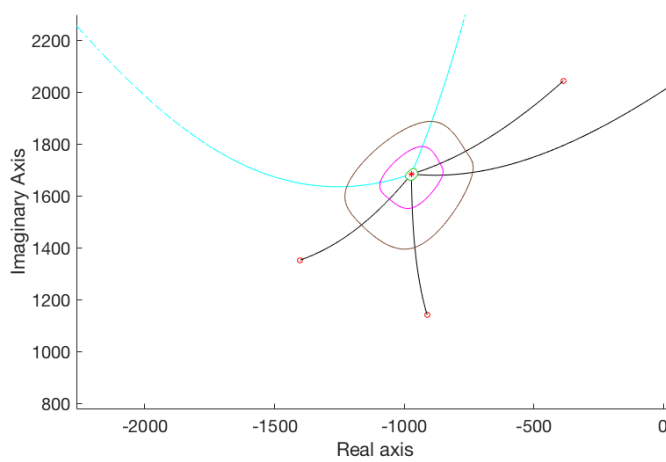
Application 2. To demonstrate how the procedure works in cases of larger matrices, in this application we examine the matrix $A = 10^{-7} \cdot \text{Pores2}$, where Pores2 is a 1024×1024 matrix from the Harwell-Boeing sparse matrix collection [22] related to a non-symmetric computational fluid dynamics problem. Here, the factor 10^{-7} is used for scaling purposes and is related to the norm- $\|\cdot\|_1$ order of the matrix under consideration. Initiating the procedure with 30 equidistributed points on the circle $\{z \in \mathbb{C} : |z| = \frac{1}{2}\|A\|_1\}$, the method required a total of 810 singular value computations for a minimum parameter value of $\epsilon_0 = 0.005$; the resulting pseudospectra visualizations for $\epsilon = 10^{-1}, 10^{-1.5}, 10^{-2}$ are depicted in Figure 2. For this example, we have opted not to introduce additional points.



(a) Trajectories of 10 sequences converging to $\sigma(A)$.



(b) Additional interior points (red circles) and relevant trajectories (solid black lines).



(c) Pseudospectrum component in the upper left side for $\epsilon = 1, 5, 10$.

Figure 1. Pseudospectrum computation for random $A \in \mathbb{C}^{6 \times 6}$ with spectral gaps, using 10 initial points.

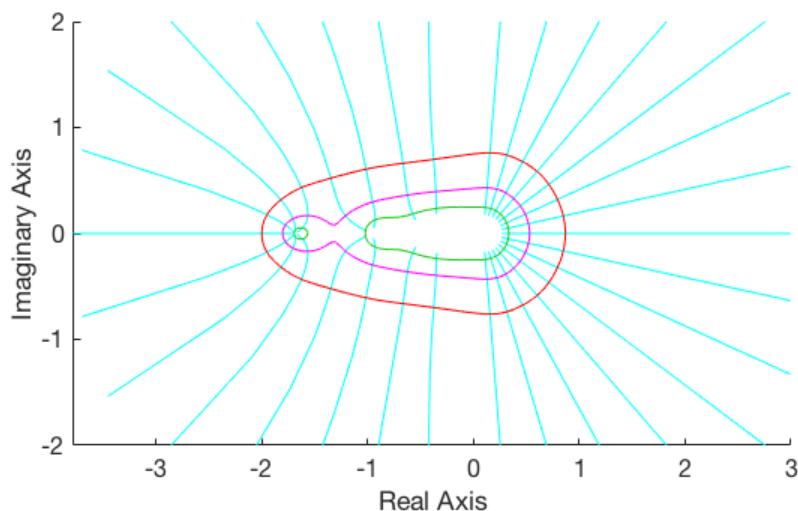


Figure 2. Pseudospectra computations for a non-symmetric sparse matrix of order 1024 from the Harwell-Boeing collection and $\epsilon = 10^{-1}, 10^{-1.5}, 10^{-2}$.

Perron root computation. Applications of non-negative matrices, i.e., matrices with exclusively non-negative real entries, abound in such diverse fields as probability theory, dynamical systems, Internet search engines, tournament matrices etc. In this context, the dominant eigenvalue of non-negative matrices, also referred to as *Perron root*, is of central importance. Localization of the Perron root has been extensively studied in the literature; relevant bounds can be found in [23–27]. Its computation is typically carried out using the power method; the convergence rate of this approach depends on the relative magnitudes of the two dominant eigenvalues. Relevant methods have appeared in [28–30], among others. As a second application of the approximating sequences (2), the following experiment reports an elegant way of approximating Perron roots.

Application 3. For this experiment, we considered a tuple of 50 non-negative matrices $\{A_\ell\}_{\ell=1}^{50} \subset \mathbb{R}_+^{500 \times 500}$ with uniformly distributed entries in $(0, 50)$. The symmetry of $\sigma_\epsilon(A_\ell)$ with respect to the real axis suggests that it suffices to restrict the computations exclusively to the closed upper half-plane. Hence, for each of the matrices A_ℓ , we initiated the construction of the sequences (2) from equidistributed initial terms $\{\mu_0^{\ell,j}\}_{j=1}^{10} \subset \{z \in \mathbb{C} : |z| = 10^4 \cdot \|A_\ell\|_1, \text{Im}(z) \geq 0\}$ ($\ell = 1, \dots, 50$). As expected, the rightmost of these points formed sequences converging to the Perron root of A_ℓ , while each of the remaining ones approximated some other peripheral eigenvalue. In the generic case, the magnitude of the second highest eigenvalue of A_ℓ was much smaller than the Perron root. Figure 3 is illustrative of this separation; the blue curve traces the boundary of the numerical range of such a matrix, red points indicate its eigenvalues, while the cyan lines correspond to the trajectories of the constructed sequences. Denoting $\{\mu_k^{\ell,j}\}_{k=0}^{n_{\ell,j}}$ ($j = 1, \dots, s_\ell$) those sequences approximating the Perron root $\lambda_\ell \in \sigma(A_\ell)$ ($\ell = 1, \dots, 50$), then the relative error in each iteration $\frac{|\mu_k^{\ell,j} - \lambda_\ell|}{|\lambda_\ell|}$, $k = 0, 1, \dots, \min(n_{\ell,j})$, decreases rapidly, even though the initial points $\mu_0^{\ell,j}$ ($j = 1, \dots, s_\ell$) were chosen to be extremely remote from $\sigma(A_\ell)$. Averages

$$\frac{1}{\ell} \sum_{\ell=1}^{50} \frac{1}{s_\ell} \sum_{j=1}^{s_\ell} \frac{|\mu_k^{\ell,j} - \lambda_\ell|}{|\lambda_\ell|}$$

of these relative approximation errors over the tuple of matrices for the first $k = 1, 2, \dots, 5$ iterations are demonstrated in the first column of Table 1, verifying that a reliable estimate for the Perron root may in the generic case be obtained after the computation of as few as 3 terms in the corresponding trajectories.

The remaining $(10 - s_\ell)$ sequences converge to some other peripheral eigenvalues $\lambda_{\ell,1}, \dots, \lambda_{\ell,s_\ell} \in \sigma(A_\ell)$, reasonable approximations of which require a rather larger number of iterations, as can be seen from the second column of Table 1 reporting.

$$\frac{1}{\ell} \sum_{k=1}^{50} \frac{1}{10 - s_\ell} \sum_{j=s_\ell+1}^{10} \frac{|\mu_k^{\ell,j} - \lambda_{\ell,j}|}{|\lambda_{\ell,j}|}.$$

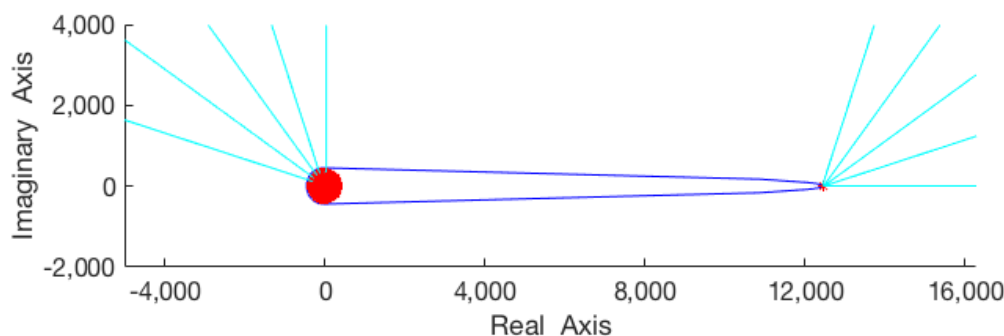


Figure 3. Indicative numerical range of 500×500 non-negative matrix and 10 approximating trajectories.

Table 1. Relative approximation errors for Perron root and other peripheral eigenvalues of 500×500 non-negative matrices.

# of Iterations	Mean Rel. Error (Perron Root)	Mean Rel. Error (Other Eigenvalues)
1	0.0011	0.4205
2	7.0082×10^{-7}	0.1783
3	4.4907×10^{-10}	0.1030
4	2.8798×10^{-13}	0.0680
5	9.2285×10^{-16}	0.0483

Application 3 suggests that any reasonable upper bound $\mu_0 \in \mathbb{R}$ suffices to yield reliable estimations for the Perron root after computation of only 2–3 terms in the sequence (2).

The previous experiment may seem excessively optimistic. Indeed, there can be instances when the situation is much more demanding.

Application 4. The Frank matrix is well-known to have ill-conditioned eigenvalues. For this application, we test the behavior of the proposed method on the Frank matrix of order 32, the normalized matrix of eigenvectors of which has condition number 7.81×10^{11} . Figure 4 depicts the resulting pseudospectra visualizations for $\epsilon = 0.001, 0.005, 0.01, 0.02, 0.03$, initiating the procedure from 30 points located on the upper semiellipse centered at $(40, 0)$ with semi-major and semi-minor axes lengths equal to 70 and 15, respectively. The depicted trajectories were constructed, so that the final terms in each polygonal chain lie within $\sigma_{0.001}(A)$. Then, according to the distances of the final terms of consecutive sequences, at most two additional points are introduced on the line segment connecting these respective final terms. The necessary iterations for the construction of the relevant sequences are reported in Table 2 for different numbers of initial points.

The approximating quality of the sequences is much compromised when compared to the generic case, requiring many more iterations, especially for the eigenvalues with smallest real parts; these are also the most ill-conditioned ones. In fact, the seven rightmost sequences converging to the Perron root (refer to Figure 4) display the fastest convergence, the second group of thirteen sequences leading to the intermediate eigenvalues being somewhat more compromised, while the leftmost sequences naturally exhibit even more diminished approximation quality. Mean relative approximation errors for these three groups are reported in Table 3.

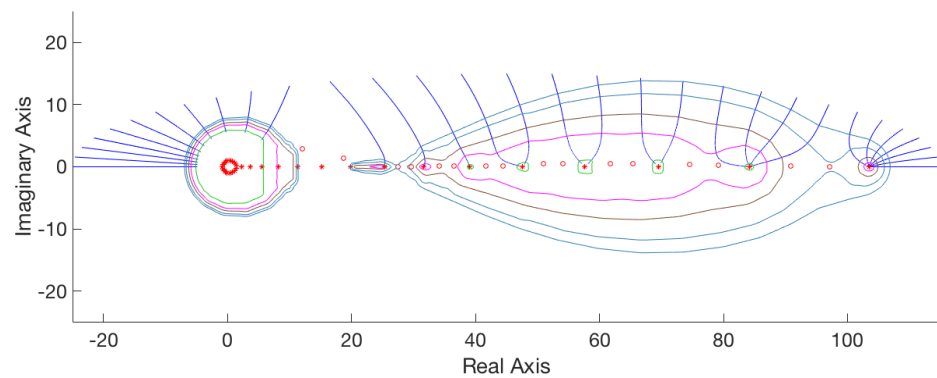


Figure 4. Pseudospectra computation for the Frank matrix of order 32 and $\epsilon = 0.001, 0.005, 0.01, 0.02, 0.03$. Additional points selected between the endpoints of the initial sequences are denoted by red circles, while eigenvalues are denoted by red stars.

Table 2. Number of iterations for different numbers of initial points ($\epsilon_0 = 0.001$).

# of Initial Points	10	15	30
Iterations (initial points)	11,206	18,455	35,159
Iterations (additional points)	16,116	14,872	11,883
Iterations (total)	27,322	33,327	47,042

Table 3. Relative approximation errors for Perron root and other eigenvalues of the Frank matrix of order 32.

# of Iterations	Mean Rel. Error (Perron Root)	Mean Rel. Error (Intermediate Eigenvalues)	Mean Rel. Error (Leftmost Eigenvalues)
1	0.0916	0.2795	11.0866
100	0.0373	0.1507	5.7968
200	0.0192	0.1206	5.3541
300	0.0103	0.1103	5.1174
400	0.0055	0.0956	4.9582
500	0.0029	0.0843	4.8652

For the numerical experiments in this section, we have restricted ourselves to initial points encircling the spectrum. Another option would be to use our method in tandem with randomization techniques for the initial points selection.

3. Matrix Polynomials

The derivation of eigenvalue approximating sequences may be readily extended to account for the general matrix polynomial case

$$P(\lambda) = A_m\lambda^m + A_{m-1}\lambda^{m-1} + \dots + A_1\lambda + A_0,$$

where $\lambda \in \mathbb{C}$ and $A_j \in \mathbb{C}^{n \times n}$ ($j = 0, 1, \dots, m$), with $A_m \neq 0$. Recall that the spectrum of $P(\lambda)$ is the set of all its eigenvalues; i.e., $\sigma(P) = \{\lambda \in \mathbb{C} : \det P(\lambda) = 0\}$. For a scalar $\lambda_0 \in \sigma(P)$, the nonzero solutions $v_0 \in \mathbb{C}^n$ to the system $P(\lambda_0)v_0 = 0$ are the eigenvectors of $P(\lambda)$ corresponding to λ_0 .

The ϵ -pseudospectrum of $P(\lambda)$ was introduced in [14] for a given parameter $\epsilon > 0$ and a set of nonnegative weights $\mathbf{w} \in \mathbb{R}_+^{m+1}$ as the set

$$\sigma_{\epsilon, \mathbf{w}}(P) = \{\lambda \in \mathbb{C} : \det P_\Delta(\lambda) = 0, \|\Delta_j\| \leq \epsilon w_j, j = 0, 1, \dots, m\} \tag{3}$$

of eigenvalues of all admissible perturbations $P_\Delta(\lambda)$ of $P(\lambda)$ of the form

$$P_\Delta(\lambda) = (A_m + \Delta_m)\lambda^m + (A_{m-1} + \Delta_{m-1})\lambda^{m-1} + \dots + (A_1 + \Delta_1)\lambda + (A_0 + \Delta_0),$$

where the norms of the matrices $\Delta_j \in \mathbb{C}^{n \times n}$ ($j = 0, 1, \dots, m$) satisfy the specified (ϵ, \mathbf{w}) -related constraints. In contrast to the constant matrix case, a whole tuple of perturbing matrices Δ_j is involved, which explains the presence of the additional parameter vector \mathbf{w} in the definition of $\sigma_{\epsilon, \mathbf{w}}(P)$. However, considering for some $A \in \mathbb{C}^{n \times n}$ the pencil $P(\lambda) = I_n\lambda - A$, note that (3) reduces to the usual ϵ -pseudospectrum of the matrix $A \in \mathbb{C}^{n \times n}$ for the choice of $\mathbf{w} = \{w_0, w_1\} = \{1, 0\}$, since

$$\sigma_{\epsilon, \{1, 0\}}(P) = \{\lambda \in \mathbb{C} : \det(I_n\lambda - (A + \Delta_0)) = 0, \|\Delta_0\| \leq \epsilon\} = \sigma_\epsilon(A).$$

In the general case, the nonnegative weights $\{w_j\}_{j=0}^m$ allow freedom in how perturbations are measured; for example, in an absolute sense when $w_0 = w_1 = \dots = w_m = 1$, or in a relative sense when $w_j = \|A_j\|$ ($j = 0, 1, \dots, m$). On the other hand, the choice $\epsilon = 0$ leads to $\sigma_{0, \mathbf{w}}(P) = \sigma(P)$.

From a computational viewpoint, a more convenient characterization [14] (Lemma 2.1) for this set is given by

$$\sigma_{\epsilon, \mathbf{w}}(P) = \{\lambda \in \mathbb{C} : s_{\min}(P(\lambda)) \leq \epsilon q_{\mathbf{w}}(|\lambda|)\},$$

where $s_{\min}(P(\lambda))$ is the minimum singular value of the matrix $P(\lambda)$ and the scalar polynomial

$$q_{\mathbf{w}}(\lambda) = w_m\lambda^m + w_{m-1}\lambda^{m-1} + \dots + w_1\lambda + w_0,$$

is defined in terms of the weights $\{w_j\}_{j=0}^m$ used in the definition (3) of $\sigma_{\epsilon, \mathbf{w}}(P)$. In fact, since the eigenvalues of $P_\Delta(\lambda)$ are continuous with respect to the entries of its coefficient matrices, the boundary of $\sigma_{\epsilon, \mathbf{w}}(P)$ is expressed as

$$\partial\sigma_{\epsilon, \mathbf{w}}(P) \subseteq \{\lambda \in \mathbb{C} : s_{\min}(P(\lambda)) = \epsilon q_{\mathbf{w}}(|\lambda|)\}; \tag{4}$$

the equality $s_{\min}(P(\lambda)) = \epsilon q_{\mathbf{w}}(|\lambda|)$ is satisfied for some $\epsilon > 0$ only for a finite number of points $\lambda \in \text{int}(\sigma_{\epsilon, \mathbf{w}}(P))$.

Suppose now that we want to approximate an eigenvalue of a matrix polynomial which lies in the neighborhood of some point of interest $\mu_0 \in \mathbb{C} \setminus \sigma(P)$ on the complex plane. Expression (4) suggests that the derivation of a convergent sequence in Section 2 may be readily adapted for our purposes. Indeed, for every scalar $\mu_0 \in \mathbb{C}$, there exists some $\hat{\epsilon}_1 > 0$, such that $\mu_0 \in \partial\sigma_{\hat{\epsilon}_1, \mathbf{w}}$ and then (4) implies $\hat{\epsilon}_1 = \frac{s_{\min}(P(\mu_0))}{q_{\mathbf{w}}(|\mu_0|)}$. Moreover, assuming $s_{\min}(P(\mu_0))$ is a simple singular value of the matrix $P(\mu_0)$, we may invoke Theorem 1 to conclude that the function $g_P : \mathbb{C} \rightarrow \mathbb{R}_+$ with $g_P(z) = s_{\min}(P(z))$ is real analytic in a neighborhood of $\mu_0 = x_0 + iy_0$. In fact,

$$\nabla g_P(x_0 + iy_0) = \left(\text{Re} \left(u_{\min}^* \frac{\partial P(x_0 + iy_0)}{\partial x} v_{\min} \right), \text{Re} \left(u_{\min}^* \frac{\partial P(x_0 + iy_0)}{\partial y} v_{\min} \right) \right),$$

where u_{\min} and v_{\min} denote the left and right singular vectors of $P(\mu_0)$ associated to $s_{\min}(P(\mu_0)) = \hat{\epsilon}_1 q_{\mathbf{w}}(|\mu_0|)$, respectively [13] (Corollary 4.2).

As in the constant matrix case, moving from the initial point $\mu_0 \in \partial\sigma_{\hat{\epsilon}_1, \mathbf{w}}$ towards the interior of $\sigma_{\hat{\epsilon}_1, \mathbf{w}}$ in the normal direction to the curve $\partial\sigma_{\hat{\epsilon}_1, \mathbf{w}}$, the scalar

$$\mu_1 = \mu_0 - \hat{\epsilon}_1 \cdot \frac{\nabla [s_{\min}(P(z)) - \hat{\epsilon}_1 q_{\mathbf{w}}(|z|)]}{|\nabla [s_{\min}(P(z)) - \hat{\epsilon}_1 q_{\mathbf{w}}(|z|)]|} \Bigg|_{z=x_0+iy_0}$$

with $\hat{\epsilon}_1 = \frac{s_{\min}(P(\mu_0))}{q_{\mathbf{w}}(|\mu_0|)}$ can be considered to be an estimate of some eigenvalue $\lambda \in \sigma(P)$. In this way, a convergent sequence $\{\mu_k\}_{k \in \mathbb{N}}$ to the eigenvalue $\lambda \in \sigma(P)$ is recursively defined with initial point μ_0 and general term

$$\mu_k = \mu_{k-1} - \frac{s_{\min}(P(\mu_{k-1}))}{q_{\mathbf{w}}(|\mu_{k-1}|)} \cdot \frac{\nabla[s_{\min}(P(z)) - \hat{\epsilon}_{k-1}q_{\mathbf{w}}(|z|)]}{|\nabla[s_{\min}(P(z)) - \hat{\epsilon}_{k-1}q_{\mathbf{w}}(|z|)]|} \Bigg|_{z=\mu_{k-1}=x_{k-1}+iy_{k-1}} \quad (5)$$

Numerical Experiments

The steps outlined in Section 2.1.1 are readily modified using the sequences in (5) to yield spectral enclosures for matrix polynomials.

Application 5 ([31], Example 3). We consider the 50×50 matrix polynomial $P(\lambda) = A_2\lambda^2 + A_1\lambda + A_0$, where

$$A_2 = I_{50}, \quad A_1 = \text{tridiag}\{-3, 9, -3\}, \quad A_0 = \text{tridiag}\{-5, 15, -5\},$$

describing a damped mass-spring system [14,32] and set non-negative weights $\mathbf{w} = \{1, 1, 1\}$, measuring perturbations of the coefficient matrices $\{A_j\}_{j=0}^2$ in an absolute sense. We initiate the procedure with 15 equidistributed initial points $\{\mu_0^j\}_{j=1}^{15}$ on the semicircle

$$\{z \in \mathbb{C} : |z| = 15 \left(= \text{median}_{j=0,1,2}(\|A_j\|_1) \right), \text{Im}(z) \geq 0\}$$

and proceed to determine eigenvalue approximating sequences $\{\mu_k^j\}_{k=0}^{n_j}$ ($j = 1, \dots, 15$) according to (5), so that their final terms all lie in the interior of $\sigma_{0.01, \mathbf{w}}(P)$. This computation requires 722 iterations. As in the constant matrix case, interpolation between the values of ϵ_k^j such that $\mu_k^j \in \partial\sigma_{\epsilon_k^j, \mathbf{w}}(P)$ along the trajectories formed by $\{\mu_k^j\}_{k=0}^{n_j}$ ($j = 1, \dots, 15$) results in approximations of $\partial\sigma_{\epsilon, \mathbf{w}}(P)$ for $\epsilon = 0.1, 0.2, 0.3, 0.4, 0.5$, as seen in Figure 5a. Note this yields a sufficiently accurate sketch of $\sigma_{\epsilon, \mathbf{w}}(P)$ and is very competitive when compared to other methods. For instance, Figure 5b is obtained via the procedure in [31] applied to a 400×400 grid on the relevant region $\Omega = [-20, 5] \times [-15, 15]$. This latter approach is far more computationally intensive, requiring 71,575 iterations to visualize $\sigma_{\epsilon, \mathbf{w}}(P)$ for the same tuple of parameters.

In case a more detailed spectral localization is desired, our method may be adapted, as in Application 1, to identify individual pseudospectrum components. Our next experiment also serves to illustrate the fact that the number of initial trajectories that are attracted by the individual eigenvalues to form the related clusters is intimately connected to eigenvalue sensitivity.

Application 6 ([13], Example 5.1). We consider the mass-spring system from ([13], Ex. 5.2) defining the 3×3 selfadjoint matrix polynomial

$$P(\lambda) = A_2\lambda^2 + A_1\lambda + A_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 5 \end{bmatrix} \lambda^2 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 3 & -1 \\ 0 & -1 & 6 \end{bmatrix} \lambda + \begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

and set $\mathbf{w} = \{1, 1, 1\}$. As in Application 5, computations are restricted exclusively to the closed upper half-plane. However, the close proximity of the eigenvalues $-0.08 \pm i1.45, -0.75 \pm i0.86, -0.51 \pm i1.25$ (indicated by “*” in Figure 6), as well as the fact that the pair $\lambda = -0.51 \pm i1.25$ is less sensitive than the other two, necessitates the use of many initial points. Indeed, as demonstrated in Figure 6a, initiating the procedure with 40 equidistributed initial points on

$\{z \in \mathbb{C} : |z| = \min \|A_j\|_1, \text{Im}(z) \geq 0\}$ results in $\sigma(P)$ being under-represented in the resulting clusters. In order to correctly approximate all three elements of the spectrum on the upper half-plane enforces the use of as many as 80 points on the selected semicircle. The length n_j of each sequence $\{\mu_k^j\}_{k=0}^{n_j}$ ($j = 1, \dots, 80$) is determined, so that all values $\{s_{\min}(P(\mu_{n_j}^j))\}_{j=1}^{80}$ do not exceed the prefixed parameter value of $\epsilon_0 = 0.01$; this construction involved 1162 singular value computations. Using the squared Euclidean distance as the metric for computing the cluster evaluation criterion, three distinct groups are correctly identified, each converging to a different eigenvalue in the closed upper half-plane, as in Figure 6b. Note that the least sensitive eigenvalue $\lambda = -0.51 + i1.25$ ends up attracting only one of these sequences; the corresponding group being a singleton. To correctly sketch the boundaries of $\sigma_{\epsilon, \mathbf{w}}(P)$ for the triple of $\epsilon = 0.24, 0.48, 0.73$ ($> \epsilon_0 = 0.01$), we introduce six additional points for each cluster. Indeed, denoting c_1, c_2, c_3 the centroids of the clusters, for each cluster $j = 1, 2, 3$ we consider the vertices $\{p_i^j\}_{i=1}^6$ of a canonical hexagon centered at c_j with maximal diameter equal to $\min(|c_j - c_i|)_{i \neq j}$. These vertices are indicated by circles in Figure 6b and are used as starting points to construct the additional trajectories indicated by the black lines in Figure 6c. Note that all three selected parameters $\epsilon = 0.24, 0.48, 0.73$ should be possible to interpolate along these additional lines as well, which explains why most of these trajectories have been extended to the opposite directions as well, modifying the definition of the sequences in (5) in each instance accordingly. The construction of the additional sequences requires 202 singular value computations (leading to a total of 1364 iterations), while the resulting approximations of pseudospectra boundaries for $\epsilon = 0.24, 0.48, 0.73$ are depicted in Figure 6c.

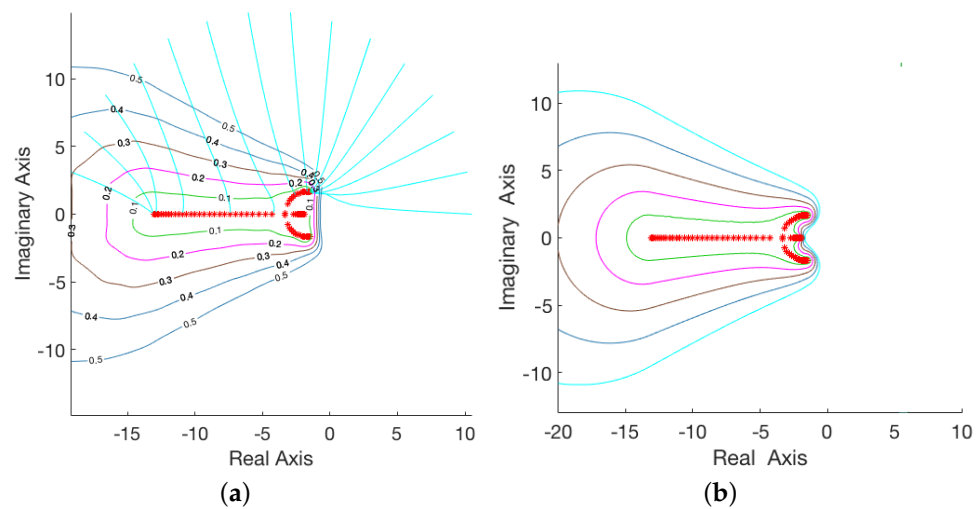
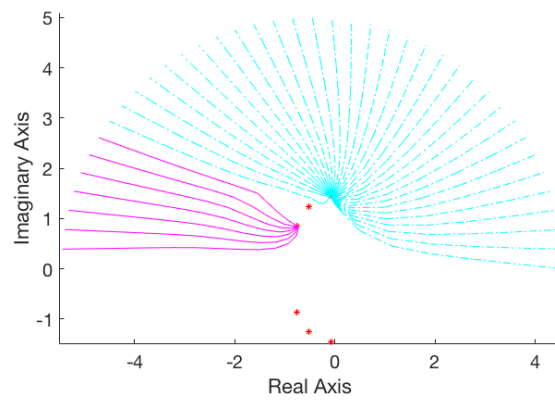
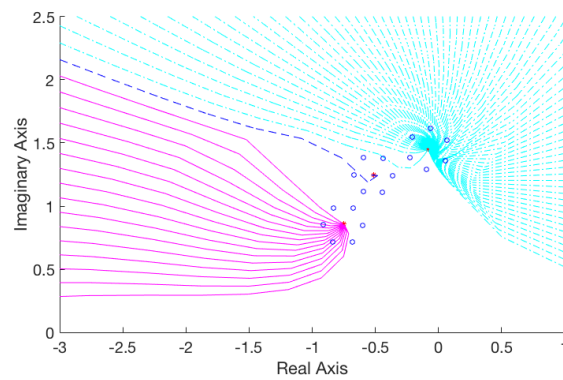


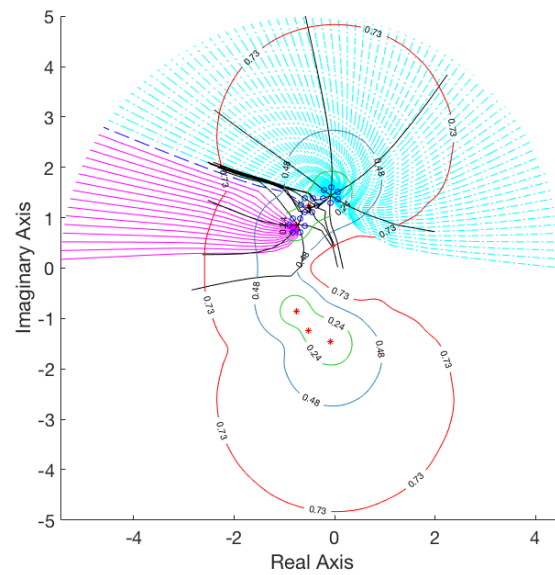
Figure 5. Pseudospectrum computation for a damped mass-spring system. (a) Approximate pseudospectra visualization, interpolating along 15 trajectories of converging sequences. (b) Pseudospectra visualization, using the modified grid algorithm in [31].



(a) Partial spectral identification, due to close eigenvalue proximity.



(b) Complete spectral identification with increased number of initial points.



(c) Pseudospectra visualizations for $\epsilon = 0.24, 0.48, 0.73$.

Figure 6. Pseudospectra computations for a vibrating system.

Application 7 ([13], Example 5.3). This experiment tests the behavior of the method on a damped gyroscopic system described by the 100×100 matrix polynomial

$$P(\lambda) = M\lambda^2 + (G + D)\lambda + K,$$

with

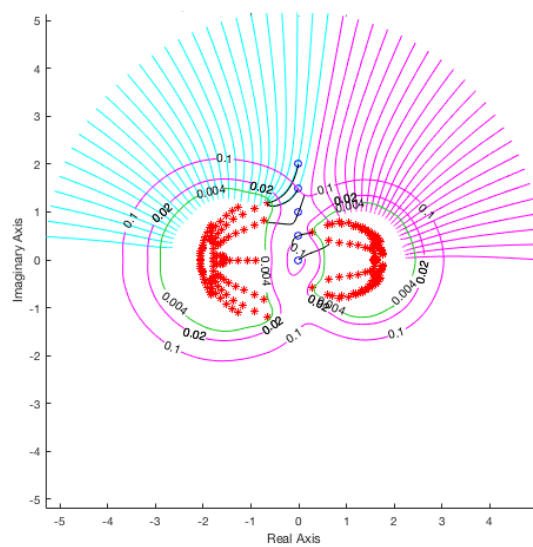
$$\begin{aligned}
 M &= I_{10} \otimes \frac{4I_{10} + B + B^T}{6} + 1.30 \frac{4I_{10} + B + B^T}{6} \otimes I_{10}, \\
 G &= 1.35I_{10} \otimes (B - B^T) + 1.10(B - B^T) \otimes I_{10}, \\
 D &= \text{tridiag}\{-0.1, 0.3, -0.1\}, \\
 K &= I_{10} \otimes (B + B^T - 2I_{10}) + 1.20(B + B^T - 2I_{10}) \otimes I_{10}
 \end{aligned}$$

and B the 10×10 nilpotent matrix having ones on its subdiagonal and zeros elsewhere. Note M, K are positive and negative definite respectively, G is skew-symmetric, and the tridiagonal D is a damping matrix.

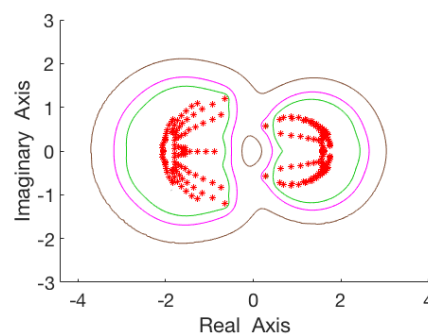
Starting with 50 points on

$$\{z \in \mathbb{C} : |z| = 15(= \text{median}(\|K\|_1, \|G + D\|_1, \|M\|_1)), \text{Im}(z) \geq 0\}$$

and then 5 additional points on the perpendicular bisector of the line segment defined by the two centroids of the resulting clusters (indicated by the blue circles), the resulting pseudospectrum approximation required 1212 iterations in total with $\epsilon_0 = 0.002$ and can be seen in the left part of Figure 7a. The algorithm in [31] applied to a 300×300 grid on the region $\Omega = [-4, 4] \times [-3, 3]$ required 29,110 iterations for pseudospectra visualization for the same triple $\epsilon = 0.004, 0.02, 0.1$ to obtain comparable results in Figure 7b.



(a)



(b)

Figure 7. Comparison of pseudospectra computation for a damped gyroscopic system and $\epsilon = 0.004, 0.02, 0.1$. (a) Computation using 50 initial points. (b) Computation using algorithm in [31].

We conclude this section, examining the behavior of the method on a non-symmetric example.

Application 8 ([31], Example 2). We consider the 20×20 gyroscopic system

$$P(\lambda) = A_2\lambda^2 + A_1\lambda + A_0 = I_{20}\lambda^2 + \mathbf{i} \begin{bmatrix} I_{10} & 0 \\ 0 & 5I_{10} \end{bmatrix} \lambda + \begin{bmatrix} 1 & -1 & -1 & \dots & -1 \\ -1 & 1 & -1 & \dots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \dots & 1 \end{bmatrix}$$

and $\mathbf{w} = \{1, 1, 1\}$. Starting with 21 points on

$$\{z \in \mathbb{C} : |z| = 25 (= \|A_0\|_1 + \|A_1\|_1), \text{Re}(z) \leq 0\}$$

and $\epsilon_0 = 0.001$, five clusters are detected (Figure 8a) after 1140 iterations. Then, two additional points are introduced on each of the line segments defined by the centroids of the detected clusters (indicated by the blue circles in Figure 8b), causing the iterations to rise to the total number of 2662 in order to determine the 20 corresponding trajectories (indicated by grey lines in Figure 8c). The corresponding visualizations in Figure 8d), obtained via [31] applied to a 400×400 grid on the region $\Omega = [-20, 20] \times [-15, 10]$ required 88,462 iterations.

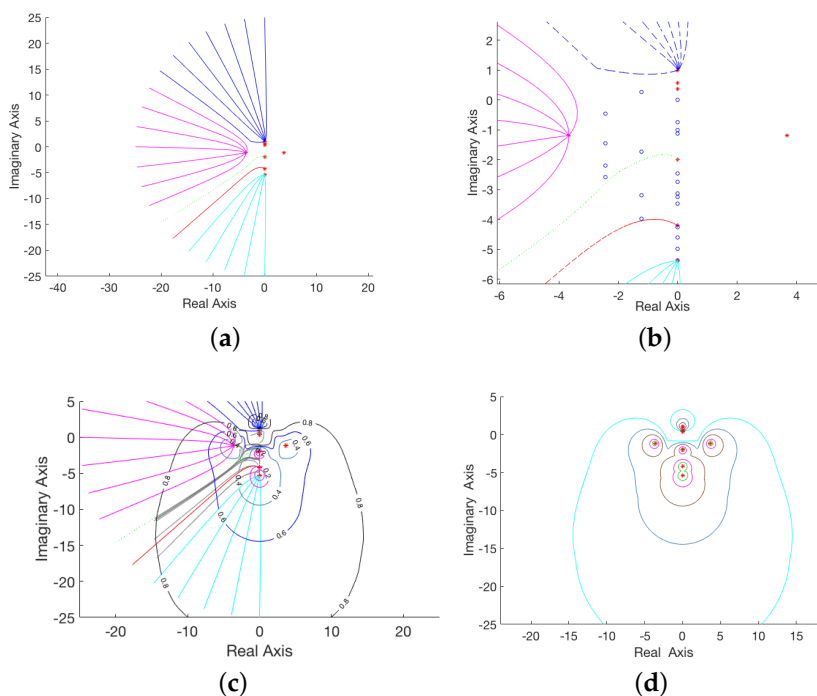


Figure 8. Comparison of pseudospectra computation for a gyroscopic system and $\epsilon = 0.1, 0.2, 0.4, 0.6, 0.8$. (a) Cluster detection using 21 initial points. (b) Locations of additional points. (c) Pseudospectra visualizations interpolating along the trajectories of 21 initial and 20 additional points. (d) Pseudospectra visualization, using the modified grid algorithm in [31].

4. Concluding Remarks

In this note, we have shown how to define sequences which, beginning from arbitrary complex scalars, converge to some element of the spectrum of a matrix. This approach can be applied both to constant matrices and to more general matrix polynomials and can be used as a means to obtain estimates for those eigenvalues that lie in the vicinity of the initial term of the sequence. This construction is also useful when no information on the location of the spectrum is a priori known. In such cases, repeating the construction from arbitrary points, it is possible to detect peripheral eigenvalues, localize the spectrum and even obtain spectral enclosures.

As an application, in this paper we used this construction to compute the pseudospectrum of a matrix or a matrix polynomial. Thus, a useful technique for speeding up

pseudospectra computations emerges, which is essential for applications. An advantage of the proposed approach is that does not make any assumptions on the location of the spectrum. The fact that all computations are performed on some dynamically chosen locations on the complex plane which converge to the eigenvalues, rather than on a large number of predefined points on a rigid grid, can be seen as improvement over conventional grid algorithms.

Parallel implementation of the method can lead to further computational savings when applied to large matrices. Another option would be to apply this method combined with randomization techniques for the selection of the initial points of the sequences. In the large-scale matrix case, this method may be helpful in obtaining a first impression of the shape and size of pseudospectrum and even computing a rough approximation. Then, if desired, this could be used in conjunction with local versions of the grid algorithm and small local meshes about individual areas of interest.

Author Contributions: All authors have equally contributed to the conceptualization of this paper, to software implementation and to the original draft preparation. Funding acquisition and project administration: P.P. All authors have read and agreed to the submitted version of the manuscript.

Funding: This research is carried out/funded in the context of the project “Approximation algorithms and randomized methods for large-scale problems in computational linear algebra” (MIS 5049095) under the call for proposals “Researchers’ support with an emphasis on young researchers–2nd Cycle.” The project is co-financed by Greece and the European Union (European Social Fund—ESF) by the Operational Programme Human Resources Development, Education and Lifelong Learning 2014–2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Landau, H.J. On Szegő’s eigenvalue distribution theorem and non-Hermitian kernels. *J. Analyse Math.* **1975**, *28*, 216–222. [[CrossRef](#)]
- Varah, J.M. On the separation of two matrices. *SIAM J. Numer. Anal.* **1979**, *16*, 216–222. [[CrossRef](#)]
- Wilkinson, J.H. Sensitivity of eigenvalues II. *Utilitas Math.* **1986**, *30*, 243–286.
- Demmel, W. A counterexample for two conjectures about stability. *IEEE Trans. Aut. Control* **1987**, *32*, 340–342. [[CrossRef](#)]
- Trefethen, L.N. Approximation theory and numerical linear algebra. In *Algorithms for Approximation, II (Shrivenham, 1988)*; Chapman & Hall: London, UK, 1990; pp. 336–360.
- Nachtigal, N.M.; Reddy, S.C.; Trefethen, L.N. How fast are nonsymmetric matrix iterations? *SIAM J. Matrix. Anal.* **1992**, *13*, 778–795. [[CrossRef](#)]
- Mosier, R.G. Root neighborhoods of a polynomial. *Math. Comput.* **1986**, *47*, 265–273. [[CrossRef](#)]
- Reddy, S.C.; Trefethen, L.N. Lax-stability of fully discrete spectral methods via stability regions and pseudo-eigenvalues. *Comput. Methods Appl. Mech. Eng.* **1990**, *80*, 147–164. [[CrossRef](#)]
- Higham, D.J.; Trefethen, L.N. Stiffness of ODEs. *BIT Numer. Math.* **1993**, *33*, 285–303. [[CrossRef](#)]
- Davies, E.; Simon, B. Eigenvalue estimates for non-normal matrices and the zeros of random orthogonal polynomials on the unit circle. *J. Approx. Theory* **2006**, *141*, 189–213. [[CrossRef](#)]
- Böttcher, A.; Grudsky, S.M. *Spectral Properties of Banded Toeplitz Matrices*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2005.
- Van Dorsselaer, J.L.M. Pseudospectra for matrix pencils and stability of equilibria. *BIT Numer. Math.* **1997**, *37*, 833–845. [[CrossRef](#)]
- Lancaster, P.; Psarrakos, P. On the pseudospectra of matrix polynomials. *SIAM J. Matrix Anal. Appl.* **2005**, *27*, 115–129. [[CrossRef](#)]
- Tisseur, F.; Higham, N.J. Structured pseudospectra for polynomial eigenvalue problems with applications. *SIAM J. Matrix Anal. Appl.* **2001**, *23*, 187–208. [[CrossRef](#)]
- Trefethen, L.N.; Embree, M. *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*; Princeton University Press: Princeton, NJ, USA, 2005.
- Brühl, M. A curve tracing algorithm for computing the pseudospectrum. *BIT* **1996**, *36*, 441–454. [[CrossRef](#)]
- Sun, J.-G. A note on simple non-zero singular values. *J. Comput. Math.* **1988**, *62*, 235–267.
- Kokioyopoulou, E.; Bekas, C.; Gallopoulos, E. Computing smallest singular triples with implicitly restarted Lanczos biorthogonalization. *Appl. Num. Math.* **2005**, *49*, 39–61. [[CrossRef](#)]

19. Bekas, C.; Gallopoulos, E. Cobra: Parallel path following for computing the matrix pseudospectrum. *Parallel Comput.* **2001**, *27*, 1879–1896. [[CrossRef](#)]
20. Bekas, C.; Gallopoulos, E. Parallel computation of pseudospectra by fast descent. *Parallel Comput.* **2002**, *28*, 223–242. [[CrossRef](#)]
21. Trefethen, L.N. Computation of pseudospectra. *Acta Numer.* **1999**, *9*, 247–295. [[CrossRef](#)]
22. Duff, I.S.; Grimes, R.G.; Lewis, J.G. Sparse matrix test problems. *ACM Trans. Math. Softw.* **1989**, *15*, 1–14. [[CrossRef](#)]
23. Kolotilina, L.Y. Lower bounds for the Perron root of a nonnegative matrix. *Linear Algebra Appl.* **1993**, *180*, 133–151. [[CrossRef](#)]
24. Liu, S.L. Bounds for the greater characteristic root of a nonnegative matrix. *Linear Algebra Appl.* **1996**, *239*, 151–160. [[CrossRef](#)]
25. Duan, X.; Zhou, B. Sharp bounds on the spectral radius of a nonnegative matrix. *Linear Algebra Appl.* **2013**, *439*, 2961–2970. [[CrossRef](#)]
26. Xing, R.; Zhou, B. Sharp bounds on the spectral radius of nonnegative matrices. *Linear Algebra Appl.* **2014**, *449*, 194–209. [[CrossRef](#)]
27. Liao, P. Bounds for the Perron root of nonnegative matrices and spectral radius of iteration matrices. *Linear Algebra Appl.* **2017**, *530*, 253–265. [[CrossRef](#)]
28. Elsner, L.; Koltracht, I.; Neumann, M.; Xiao, D. On accurate computations of the Perron root. *SIAM J. Matrix. Anal.* **1993**, *14*, 456–467. [[CrossRef](#)]
29. Lu, L. Perron complement and Perron root. *Linear Algebra Appl.* **2002**, *341*, 239–248. [[CrossRef](#)]
30. Dembélé, D. A method for computing the Perron root for primitive matrices. *Numer. Linear Algebra Appl.* **2021**, *28*, e2340. [[CrossRef](#)]
31. Fatouros, S.; Psarrakos, P. An improved grid method for the computation of the pseudospectra of matrix polynomials. *Math. Comp. Model.* **2009**, *49*, 55–65. [[CrossRef](#)]
32. Tisseur, F.; Meerbergen, K. The quadratic eigenvalue problem. *SIAM Rev.* **1997**, *39*, 383–406. [[CrossRef](#)]