VISUALIZING PSEUDOSPECTRA FOR POLYNOMIAL EIGENVALUE PROBLEMS

Adéla Klimentová[*], Michael Šebek[**]

[*]Department of Control Engineering
Czech Technical University in Prague
[**]Institute of Information Theory and Automation
Czech Academy of Sciences in Prague

Abstract

The use of pseudospectra is widespread in various applications, e.g. control theory, acoustics, vibrating systems. Through pseudospectra we can gain insight into the sensitivity of the eigenvalues of a matrix to perturbations that is convenient for robust control. We have implemented in Matlab a method to visualize ε-pseudospectra for n×n polynomial matrix of degree greater than 2. We compute pseudospectrum for each point of the complex plane using transfer function approach. Although it might seem to be time consuming, we have testified that other methods as for instance curve tracing algorithm don't give good results. They show problems with convergence. We have also tested straighforward computing after the definition which was more time consuming than the transfer function approach. To visualize pseudospectra we used above all functions `contour` and `contourf.`

1. Introduction

Summarizing the history of pseudospectra in just a few words we can say that it have been invented several times matching various problems as stability of invariant subspaces of matrices, developing techniques for guaranteed-accuracy eigenvalue computations, unstable eigenvalues of spectral discretization matrices for differential matrices and structured perturbations of a matrix in context of spectral value sets in control theory.
First, the definitions of pseudospectra for numerical matrices appeared, followed by advances in polynomial eigenvalue problems. Let mention here both of them.
Consider matrix $A \in \mathbf{C}^{n \times n}$ and $\Lambda(A)$ be the spectrum of A. The smallest singular value is denoted by $\sigma_{min}(A)$ and Euclidean norm for vectors and spectral norm as the corresponding norm for matrices by $\| \cdot \|$. Than the following observation by Trefethen [1] is primary for the definition of the ε-pseudospectrum:
For $A \in \mathbf{C}^{n \times n}$, $\varepsilon \geq 0$ and $z \in \mathbf{C}$ the following conditions are equivalent:
1. $\|(zI - A)^{-1}\| \geq \varepsilon^{-1}$
2. there exists a matrix $E \in \mathbf{C}^{n \times n}$, $\|E\| \leq \varepsilon$, such that $z \in \Lambda(A + E)$
3. $\sigma^{min}(zI - A) \leq \varepsilon$

The collection $\Lambda_\varepsilon(A)$ of complex numbers z that fulfills one of the above conditions is called the ε-pseudospectrum of matrix A [2].
In polynomial case we work with matrix $P(\lambda) = \lambda^m A_m + \lambda^{m-1} A_{m-1} + \ldots + A_0$ where $A_k \in \mathbf{C}^{n \times n}$, $k = 0{:}m$. The polynomial eigenvalue problem is to find the solutions (x, λ) of $P(\lambda)x = 0$. If $x \neq 0$ then λ is called an eigenvalue and x the corresponding right eigenvector. The set of eigenvalues of P is denoted by $\Lambda(P)$.

The ε-pseudospectrum of P is defined by

$$\Lambda_\varepsilon(P) = \{\lambda \in C : (P(\lambda) + \Delta P(\lambda))x = 0 \text{ for some } x \neq 0 \text{ and } \Delta P(\lambda)$$
$$\text{with } \|\Delta A_k\| \leq \varepsilon \alpha_k, k = 0 : m\}$$
(1)

where $\Delta P(\lambda) = \lambda^m \Delta A_m + \lambda^{m-1} \Delta A_{m-1} + ... + \Delta A_0$ [3]. In (1) α_k are nonnegative parameters that allow freedom in how perturbations are measured.

When $P(\lambda) = A - \lambda I$, $\Delta P(\lambda) = \Delta A$ and $\alpha_1 = 1$, definition (1) reduces to the standard definition of ε-pseudospectrum of a single matrix

$$\Lambda_\varepsilon(A) = \{\lambda \in C : \lambda \in \Lambda(A + \Delta A) \text{ for some } \Delta A \text{ with } \|\Delta A\| \leq \varepsilon\}$$
(2)

In recent years, several methods for computing pseudospectra have been presented [2], [3], [4]. Some of these will be dealt in following paragraph.

2. Existing methods

In general, there exist two possible ways how to solve the problem of computing and vizualizing pseudospectra [5]. The first class of algorithms is represented by so called *grid algorithms*. In brief, these algorithms consist of selecting appropriate region of the complex plane, covering it with a grid and computing values of pseudospectrum for each z on the grid. In the second class belong *path-following algorithms*. In particular steps of these algorithms it is fundamental to find a point on the boundary of the desired pseudospectrum, follow from it a curve in the complex plane on which the value $\sigma^{min}(zI - A)$ is constant.

Both grid and path-following approaches have limitations. Grid algorithms require greater computing time than path-following algorithm because the minimum singular value of *zI-A* must be computed for each node of the grid. Therefore, the user of software working on this principle should be very cautious with fineness and size of the grid. On the other hand, path-following algorithms appear to be dangerous due to problems with convergence.

3. Extension for higher degree polynomial matrices

Traditional algorithms for vizualizing ε-pseudospectra of polynomial matrices are usually restricted on polynomial matrices up to quadratic ones and don't handle matrices of higher degree. To solve this problem we have decided to apply and test a transfer function approach [3] with extension on polynomial matrices of order greater than 2.

We outline here basic steps of this algorithm. Let consider the equation

$$P(z) = (z^m A_m + z^{m-1} A_{m-1} + ... + A_0)v = u$$
(3)

that can be written as

$$(F - zG)\begin{bmatrix} v \\ w_2 \\ \vdots \\ w_m \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -u \end{bmatrix}$$
(4)

where *F* and *G*, called companion matrices, are defined by

$$
F = \begin{bmatrix}
0 & I & 0 & \cdots & 0 \\
0 & 0 & I & \ddots & \vdots \\
\vdots & & & \ddots & 0 \\
& & & & I \\
-A_0 & -A_1 & -A_2 & \cdots & -A_{m-1}
\end{bmatrix}, G = \begin{bmatrix}
I & & & \\
& I & & \\
& & \ddots & \\
& & & I \\
& & & & A_m
\end{bmatrix}. \tag{5}
$$

Hence

$$
P(z)^{-1}u = v = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix}(F-zG)^{-1}\begin{bmatrix} 0 \\ \vdots \\ 0 \\ -u \end{bmatrix} = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix}(F-zG)^{-1}\begin{bmatrix} 0 \\ \vdots \\ 0 \\ -I \end{bmatrix}u \tag{6}
$$

and finally

$$
P(z)^{-1} = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix}(F-zG)^{-1}\begin{bmatrix} 0 \\ \vdots \\ 0 \\ -I \end{bmatrix} \tag{7}
$$

After these matrix operations we have to figure out the largest singular value of $P(z)^{-1}$.
This method is obviously effective even for higher degree polynomial matrices unlike e.g. methods based on Schur form that are usable only for degree m ≤ 2.
The presented method was implemented in Matlab in function `pspec`. Functions `contour` and `contourf` are used to draw the pseudospectrum. We show some of the results here.
In the first example we create a 4-by-4 polynomial matrix of degree 4

```
P = prand(4,4)
P =
  Column 1
    0.071 + 0.32s + 0.5s^2 + 1.3s^3 - 0.55s^4
    1.2 - 0.015s + 0.54s^2 - 0.72s^3 - 0.66s^4
   -0.02 + 0.28s + 1.1s^2 + 0.62s^3 - 1.8s^4
   -0.33 - 0.5s - 0.036s^2 - 0.17s^3 - 0.96s^4

  Column 2
    0.26 - 0.013s - 0.58s^2 + 2.1s^3 - 0.26s^4
    0.31 + 0.11s + 1.8s^2 - 0.28s^3 + 2.2s^4
    0.7 + 0.81s + 0.64s^2 + 1.3s^3 + 0.33s^4
    1.3 + 0.44s + 1.3s^2 - 0.5s^3 - 1.1s^4

  Column 3
   -1.4 + 1.8s + 0.33s^2 - 1.1s^3 + 0.62s^4
    1.5 - 1.9s - 1.7s^2 - 0.57s^3 - 0.19s^4
   -0.67 - 0.15s - 2.4s^2 + 0.47s^3 + 0.12s^4
    0.81 + 0.041s - 0.76s^2 - 0.089s^3 - 2s^4

  Column 4
    1.3 - 0.9s + 0.14s^2 - 0.14s^3 - 1.2s^4
    0.0089 + 0.84s - 0.72s^2 - 0.72s^3 - 0.2s^4
   -0.59 - 0.65s - 1.1s^2 - 0.048s^3 + 0.38s^4
```
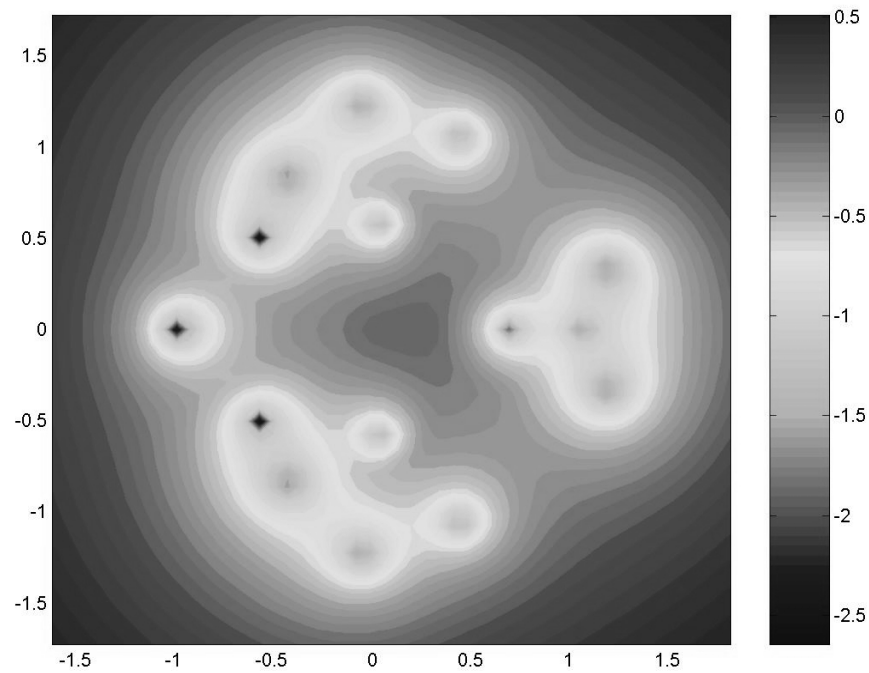
```
         1.1 - 0.98s - 0.69s^2 + 1.3s^3 - 0.91s^4
```
and vizualize its pseudospectra by typing
```
pspec(P)
```


Fig. 1 Pseudospectra of matrix P from the first example (with default settings of the grid)

In the second example we assume matrix with imaginary ceofficients:
```
P = prand(4,8) +j*prand(4,8);
```
We set some options of the vizualization by structure
```
O = struct('grid',50,'levels',50,'epsilon',[],'axis',[])
O =
        grid: 50
      levels: 50
     epsilon: []
        axis: []
```
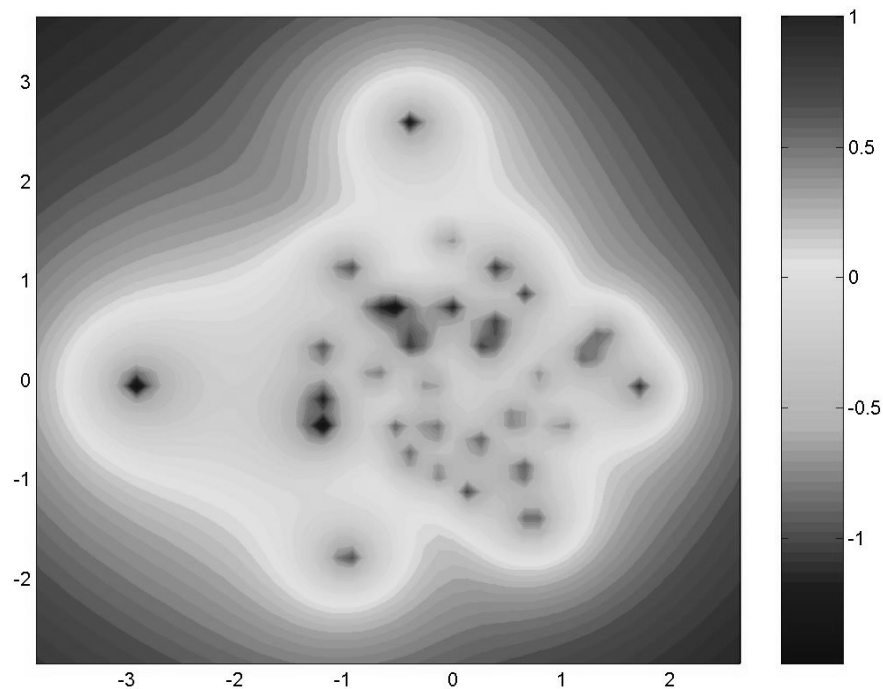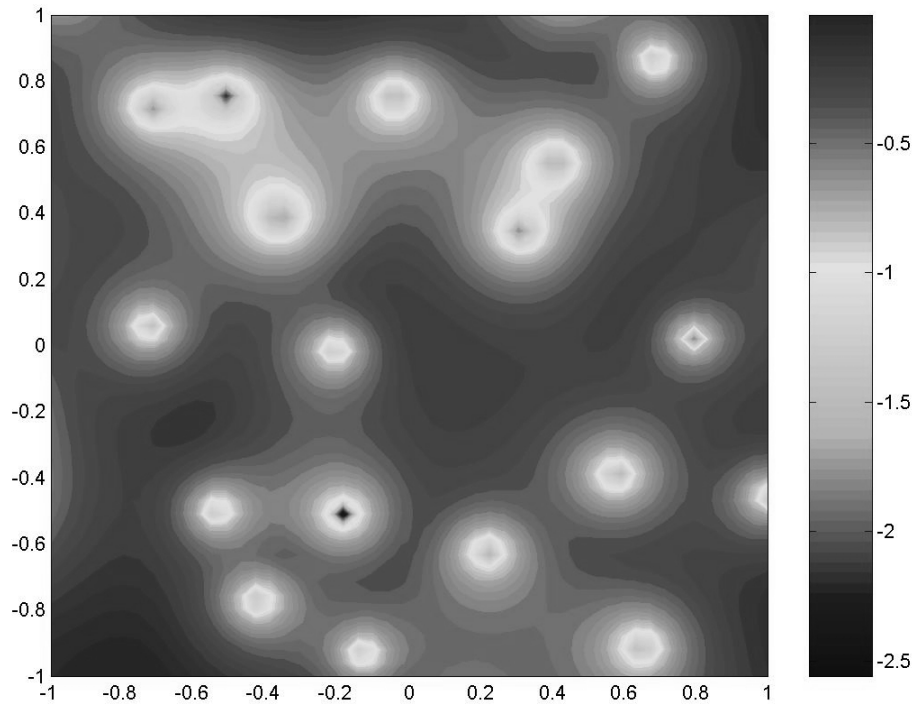Than the pseudospectra are computed by
```
pspec(P,O)
```


Fig. 2 Pseudospectra of the polynomial matrix with complex coefficients from second example

By changing the axis settings we can "cut out" another region:

```
O = struct('grid',50,'levels',50,'epsilon',[],'axis',[-1 1 -1 1])
O =

        grid: 50
      levels: 50
     epsilon: []
        axis: [-1 1 -1 1]
pspec(P,O);
```


Fig. 3 Cutout of the previous figure

There are many other possibilities how to depict the ε-pseudospectrum. Look at the example in the next paragraph.

4. Example

In MIMO systems in control theory the location of the eigenvalues of matrix polynomials determine the stability of the system.
Consider the matrix polynomial

$$P(z) = z^2 I + z \begin{bmatrix} 0 & 1+\alpha \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 0.5 & 0 \\ 0 & 0.25 \end{bmatrix} \tag{8}$$

We are searching for values of α for which $P(z)$ has all its eigenvalues inside the unit circle. To solve such a task a visualization of pseudospectra of matrix $P(z)$ can be used as shown in Fig. 4.
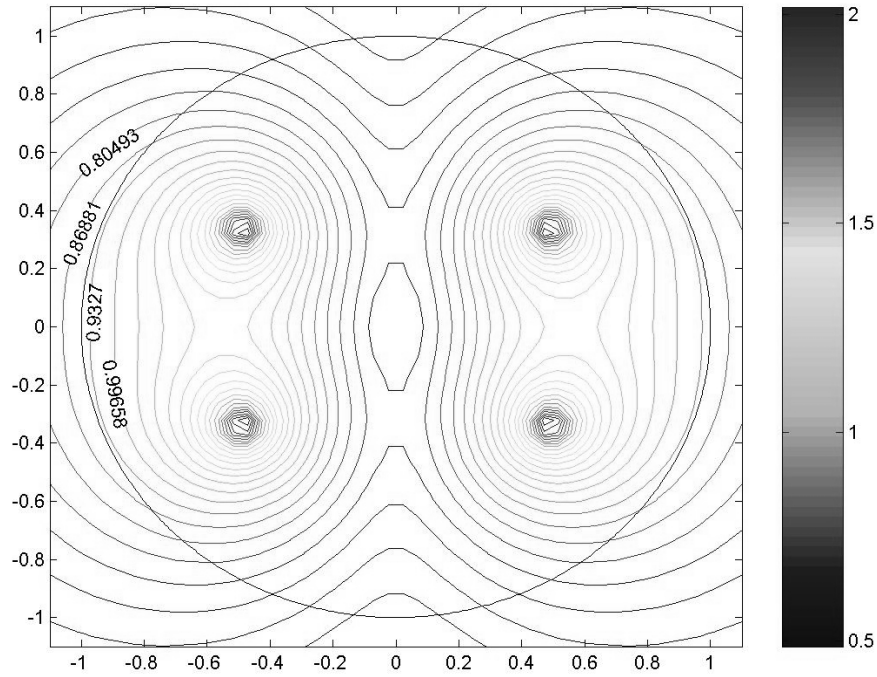
Fig. 4 From this vizualization the values of α ,for which *P(z)* has all its eigenvalues inside the unit circle, can be easily read

Here we have proposed an application of ε-pseudospectra in domain of robust control.

5. Conclusions

We have implemented and testified a transfer function approach to visualize an ε-pseudospectra of n-by-n polynomial matrix of degree even greater than 2. This algorithm seems to be suitable for this task. Computing pseudospectrum after the definition appeared to be more time consuming. We have also excluded path-following algorithms that are more convenient for numerical matrices. In case of polynomial matrices they show problems with convergence.

Further work aims at visualizing pseudospectra when eigenvalues at infinity exist. In this case, displaying pseudospectrum in 2D might be confusing. Therefore it is advisable to expand to 3D and represent pseudospectrum on Riemann sphere using stereographic projection.

6. Acknowledgement

7. References

[1] L.N.Trefethen, *Spectra and Pseudospectra: The behavior of Non-normal Matrices and Operators*, Book in preparation.
[2] M. Brühl, *A Curve Tracing Algorithm for Computing the Pseudospectrum*, 1996, BIT, 36:3, pp.441
[3] F.Tisseur, N.J. Higham, *Structured Pseudospectra for Polynomial Eigenvalue Problems, with Applications*, 2001, SIAM J. Matrix Anal. Appl., 23:1, pp.187-208
[4] N.J. Higham, F. Tisseur, *More on Pseudospectra for Polynomial Eigenvalue Problems and Applications in Control Theory*, 2002, Linear Algebra and Its Applications, 351-352, pp.435-453
[5] *Pseudospectra Gateway*, http://web.comlab.ox.ac.uk/projects/pseudospectra/index.html

Contacts

Adéla Klimentová
Department of Control Engineering
FEE CTU
Karlovo náměstí 13
121 35 Praha 2
E-mail: klimena@control.felk.cvut.cz

Michael Šebek
Institute of Information Theory and Automation
Czech Academy of Sciences in Prague
Pod vodárenskou věží 4
182 08 Praha 8 - Libeň
E-mail: msebek@utia.cas.cz