SF3584: Block 1

Bounds with eigenvalues/pseudospectra. Flexible GMRES. Left, right, preconditioners. right-hand side dependence bounds.

Block leader: Elias

Time-period: 16 Feb - 13 March

Reading material:

- Convergence bounds for iterative methods
    - Eigenvalues & pseudospectra: Book Spectra and Pseudospectra, Trefethen and Embree. Pages 12-23, pages 244-253. (Scanned pages here: https://kth.instructure.com/groups/33520/files)
    - Right-hand side dependence: Paper. GMRES convergence bounds that depend on the right-hand-side vector. David Titley-Peloquin Jennifer Pestana Andrew J. Wathen. IMA Journal of Numerical Analysis, Volume 34, Issue 2, 1 April 2014, Pages 462–479, https://doi.org/10.1093/imanum/drt025
- Preconditioning variations:
    - Left-right preconditioner, & split precond for GMRES and CG: Book by Saad Iterative methods. Section 9.2-9.3. Book available online: http://www-users.cs.umn.edu/~saad/IterMethBook_2ndEd.pdf
    - Flexible GMRES: Paper by Saad: http://epubs.siam.org/doi/abs/10.1137/0914028

Lecture material:

- Slides from course intro lecture: "Presentation lecture 1.pdf" here: https://kth.instructure.com/groups/33520/files/folder/block1
- Giampaolo notes from lecture: https://drive.google.com/open?id=14eLJHxILoyaEifGA8JpF9HOe2fRcWLs5 (Giampaolo writes: I think it could be a good idea to share our personal notes. This can help us, for example, in fixing typing errors. Do not feel forced to share your own notes and/or check mine, do it if you want to)

Edit this page

You can add problem slots by through "edit this page"

Pseudospectra & pseudospectra bounds

Problem 1-1

1. What is pseudospectra?

2. How can it be used to describe the convergence of iterative methods? (GMRES)

3. Which phase of the GMRES-iteration can (potentially) be better described with pseudo-spectra bounds than eigenvalue based bounds, the beginning or end?

Solution 1-1

1. Pseudospectra may be viewed as a generalization of eigenvalues. The ϵ-pseudospectra of a matrix A consist of all eigenvalues of matrices which are ϵ-close to A. See Problem 1-3 for two of the several equivalent definitions of pseudospectra.

2. Pseudospectra can be used to describe the convergence of iterative methods. This can be done, as by Trefethen, by using pseudospectra in combination with the Dunford-Taylor integral. For any polynomial $p_k \in \mathcal{P}_k$ and a matrix A

$$p_k(A) = \frac{1}{2\pi i}\int_\Gamma p_k(z)(zI - A)^{-1}dz.$$

For a fixed ϵ we may let $\Gamma = \Gamma_\epsilon$ be the union of contours enclosing the pseudospectrum $\sigma_\epsilon(A)$.

$$\|p_k(A)\| \leq \frac{1}{2\pi}\int_{\Gamma\epsilon} |p_k(z)|\|(zI - A)^{-1}\||d|z|.$$

By the definition of pseudospectrum in Problem 1-3 together with a contour integral bound

$$\|p_k(A)\| \leq \frac{L_\epsilon}{2\pi\epsilon} \max_{z\in\Gamma\epsilon} |p_k(z)| . \qquad (1)$$

where L_ϵ is the arc length of $\Gamma\epsilon$.

Clear!

For GMRES we have the well known result

$$\frac{\|r_k\|}{\|r_0\|} \leq \min_{p_k\in\mathcal{P}_k,\ p_k(0)=1} \|p_k(A)\| . \qquad (2)$$

Insertion of (1) into (2) gives the psedospectra bound

$$\frac{\|r_k\|}{\|r_0\|} \leq \frac{L_\epsilon}{2\pi\epsilon} \min_{p_k\in\mathcal{P}_k,\ p(0)=1} \max_{z\in\Gamma\epsilon} |p_k(z)|.$$

Comment: I saw afterwards that this is similar to one of the answers below. For further details, see below.

3. Pseudospectra bounds gives a family of bounds, one bound for each epsilon and can potentially better describe the first phase of the GMRES-iteration.

Moderator comment: Correct. I extended solution 3 slightly marked with color.

Problem 1-3

The first definition of pseudospectra: The values $z \in \mathbb{C}$ such that

$\|(z - A)^{-1}\| > \varepsilon^{-1}$

The second definition: The values $z \in \mathbb{C}$ such that z is an eigenvalue of A+E

for a matrix $E \in \mathbb{C}^{n \times n}$ such that $\|E\| \leq \varepsilon$.

1. There is a small error in the definition above. What is the error?
2. Prove that the two definitions are equivalent.

Solution 1-3

Solution by Parikshit

(1) The error lies in the second definition, where instead of the weak inequality $||E|| \leq \epsilon$, one needs to have the strong inequality $||E|| < \epsilon$. We assume as given that $\epsilon > 0$.

(2) Proof of equivalence:

Let us define $\sigma_1(A) = \{z \in \mathbb{C}, ||(z - A)^{-1}|| > \epsilon^{-1}\}$ and
$\sigma_2(A) = \{z \in C, z \in \sigma(A + E), E \in \mathbb{C}^{n \times n}, ||E|| < \epsilon\}$

Let $z \in \mathbb{C}$. We assume that $z \notin \sigma(A)$ because the equivalence is trivially satisfied in that case. Hence,

$$z \in \sigma_1(A) \implies ||(z - A)^{-1}|| = \alpha^{-1}, \quad 0 < \alpha < \epsilon$$

By definition of matrix norm, $\exists v, u \in \mathbb{C}^n, ||v|| = ||u| = 1$ such that

$$(z - A)^{-1} v = \alpha^{-1} u$$

$$\implies zu - Au = \alpha v$$

We can construct $E = \alpha v w^*$ (where $w^* u = 1$) so that $Eu = \alpha v$ and $||E|| = \alpha < \epsilon$ (If $|| \cdot ||$ is the 2-norm, then $w = u^*$. For other arbitrary norms, the existence of w is equivalent to the existence of a corresponding linear functional L on $\mathbb{C}^n$ such that $L(u) = 1$ and $||L|| = 1$, which is guaranteed by the Hahn-Banach theorem).

Hence,$\exists E \in C^{n \times n}$ and $||E|| < \epsilon$ such that $zu - Au = Eu$

$$\implies z \in \sigma(A + E), ||E|| < \epsilon$$

$$\implies z \in \sigma_2(A)$$

$$\implies \sigma_1(A) \subset \sigma_2(A) \qquad \textbf{Condition(1)}$$

Assuming $z \in \sigma_2(A)$

Nice !

$$\implies z \in \sigma(A + E), ||E|| < \epsilon$$

$$\implies \exists v \in \mathbb{C}^n, ||v|| = 1, \text{ and } (z - A)v = Ev \text{ or } v = (z - A)^{-1} Ev$$

$$\implies 1 = ||v|| \leq ||(z - A)^{-1}|| ||E|| \text{ (By Hölder's inequality)}$$

$$\implies ||(z - A)^{-1}|| > ||E||^{-1} > \epsilon^{-1}$$

$$\implies z \in \sigma_1(A) \implies \sigma_1(A) \supset \sigma_2(A) \qquad \textbf{Condition(2)}$$

By condition(1) and condition(2), we see that $\sigma_1(A) = \sigma_2(A)$ which proves equivalence of the two definitions.

Moderator comment: Correct. Nice!

Problem 1-34

Problem by Parikshit

Consider the following definition for pseudospectra provided in Trefethen and Embree,

$\sigma_\epsilon(A) = \{z \in \mathbb{C}^n, ||(z - A)v|| < \epsilon, \textbf{where } v \in \mathbb{C}^n, ||v|| = 1\}$

Prove that this definition is a consequence of the second definition of pseudospectra as defined in Problem 1-3.

Solution 1-34

By the second definition in Problem 1-3 we have that $\sigma_\epsilon(A)$ is the set of $z \in \mathbb{C} : z \in \sigma(A + E)$ for some $E \in \mathbb{C}^{N \times N}$ with $\|E\| < \epsilon$.

Suppose $A \in \mathbb{C}^{N \times N}$ and $E \in \mathbb{C}^{N \times N}$ with $\|E\| < \epsilon$. For the pseudoeigenvalues $z \in \sigma(A + E)$ and its corresponding normalized pseudoeigenvectors $0 \neq v \in \mathbb{C}^N$, $\|v\| = 1$ we have the equation

$$(A + E)v = zv \quad \Leftrightarrow \quad (z - A)v = Ev.$$

Hence

$$\|(z - A)v\| = \|Ev\| \leq \|E\| \|v\| = \|E\| < \epsilon.$$

Thus for $z \in \sigma_\epsilon(A)$ we have that $\|(z - A)v\| < \epsilon$ for some $0 \neq v \in \mathbb{C}^N$ with $\|v\| = 1$.

Moderator comment: Correct.

Problem 1-9

In the Numerical linear algebra course you gave convergence factor predictions for the matrix

```
alpha=50;
m=100; rand('state',5);
A = sprand(m,m,0.5);
A = A + alpha*speye(m); A=A/norm(A,1);
b = rand(m,1);
```

a) Visualize the pseudospectra, for some relevant epsilon-values, using pscont.m (see Problem 1-6 for installation of pscont.m)

b) For the pseudospectra in a) give convergence factor bounds and plot in a semilogy-figure. It is sufficient to visually identify disks.

Solution 1-9

Problem 1-4

Terminology and detail questions Trefethen&Embree:

1. The Ritz-values of the Arnoldi process (eigenvalues of H-matrix) can be expressed as pseudo-eigenvalues (elements of the pseudo-spectra). What is ε?

2. What is the *Ideal GMRES-problem*?

3. What is the Dunford-Taylor integral?

4. (optional, difficult?) There is a gap in the derivation of equation (26.12) from the integral equation stemming from the Dunfort-Taylor integral. Provide the missing reasoning.

Solution 1-4

Solution

(1)

After m steps of Arnoldi we get the following relation

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^T$$

Let (θ, z) be an eigenpair of H_m, in particular θ is a Ritz value. By multiplying the Arnoldi factorization by z we get

$$AV_m z = V_m H_m z + h_{m+1,m} v_{m+1} e_m^T z$$

$$AV_m z = \theta V_m z + h_{m+1,m} v_{m+1} e_m^T z$$

By using that V_m is orthogonal, i.e., $V_m^T V_m = I$, we get

$$AV_m z = \theta V_m z + h_{m+1,m} v_{m+1} e_m^T V_m^T V_m z$$

$$(A - h_{m+1,m} v_{m+1} e_m^T V_m^T) V_m z = \theta V_m z$$

The last equation tells us that θ is an eigenvalue (with eigenvector $V_m z$) of a perturbation of the matrix A. More precisely is an eigenvalue of $A + E$ with $E = h_{m+1,m} v_{m+1} e_m^T V_m^T$. By using that V_m is orthogonal we get

$$\|E\| = \|h_{m+1,m} v_{m+1} e_m^T V_m^T\| = h_{m+1,m}$$

Therefore $\theta \in \sigma_\varepsilon(A)$ with $\varepsilon = h_{m+1,m}$.

*

(2)

The GMRES minimizes (GMRES problem) the reminder $r_m = Ax_m - b$, in the sense that

$$\|r_m\| = \min_{\substack{deg(p) \leq m \\ p(0)=1}} \|p(A) r_0\|$$

Observe that $r_0 = b$. The ideal GMRES is obtained by consider the minimization problem without the reminder

$$\min_{\substack{deg(p) \leq m \\ p(0)=1}} \|p(A)\|$$

Observe that in this case, the optimization does not depend on the RHS b. Moreover we usually consider the ideal GMRES in order to have the error bounds (like the disk argument), since by using the sub-multiplicativity of the norm we have (starting from the GMRES problem)

$$\frac{\|r_m\|}{\|r_0\|} = \min_{\substack{deg(p) \leq m \\ p(0)=1}} \|p(A)\|$$

And we have derived the convergence result (except for the RHS dependence) starting from this relation.

(3)

Let $f(z)$ be a function that can be evaluated to a matrix. Then the matrix function $f(A)$ fulfills the following equality (Dunford-Taylor integral formula)

$$f(\lambda) = \frac{1}{2\pi i}\oint_{\Gamma} f(z)(zI-A)^{-1}dz$$

Where Γ is a closed curve containing the eigenvalues of A and $f(z)$ is analytic in the bounded region defined by Γ.

Observe that this is a generalization of the Cauchy's formula for matrix functions.

This result can be applied to the GMRES convergence result (ideal GMRES) as

$$\min_{\substack{deg(p)\leq m\\ p(0)=1}} \|p(A)\| = \min_{\substack{deg(p)\leq m\\ p(0)=1}} \|\frac{1}{2\pi i}\oint_{\Gamma} p(z)(zI-A)^{-1}dz\|$$

(4)

We now want to derive the pseudospectra bound (equation 26.12). We will use the following well known bound for contour integral:

$|\oint_{\Gamma} f(z)dz| \leq L \max_{z\in\Gamma} |f(z)|$

where L is the length of the curve Γ. By directly using this result together with the Dunford-Taylor integral formula with $\Gamma = \Gamma_\varepsilon$ the curve defining the pseudospectra $\sigma_\varepsilon(A)$, we obtain

$$\frac{\|r_m\|}{\|r_0\|} = \min_{\substack{deg(p)\leq m\\ p(0)=1}} \|p(A)\| = \min_{\substack{deg(p)\leq m\\ p(0)=1}} \|\frac{1}{2\pi i}\oint_{\Gamma_\varepsilon} p(z)(zI-A)^{-1}dz\|$$

$$\leq \frac{L_\varepsilon}{2\pi}\min_{\substack{deg(p)\leq m\\ p(0)=1}} \max_{\Gamma_\varepsilon} \|p(z)(zI-A)^{-1}\| \leq \frac{L_\varepsilon}{2\pi}\min_{\substack{deg(p)\leq m\\ p(0)=1}} \max_{\Gamma_\varepsilon} |p(z)| \min_{\substack{deg(p)\leq m\\ p(0)=1}} \max_{\Gamma_\varepsilon} \|(zI-A)^{-1}\|$$

By definition of pseudospectra (first definition 2.1) we have that

$\max_{\Gamma_\varepsilon} \|(zI-A)^{-1}\| = \frac{1}{\varepsilon}$

By replacing this in the previous inequality we obtain the pseudospectra bound for GMRES (26.12).

Moderator comment: Nice extensive self-contained explanation!

Problem 1-11

Visualize the eigenvalues of the K-matrix from problem 1-10 capacitor_tunable_piezo_domain_K.mat. Plot the 50 smallest and the 50 largest eigenvalues in one plot.

a) Linear system Kx=b. Why will GMRES never work for this problem without preconditioning?

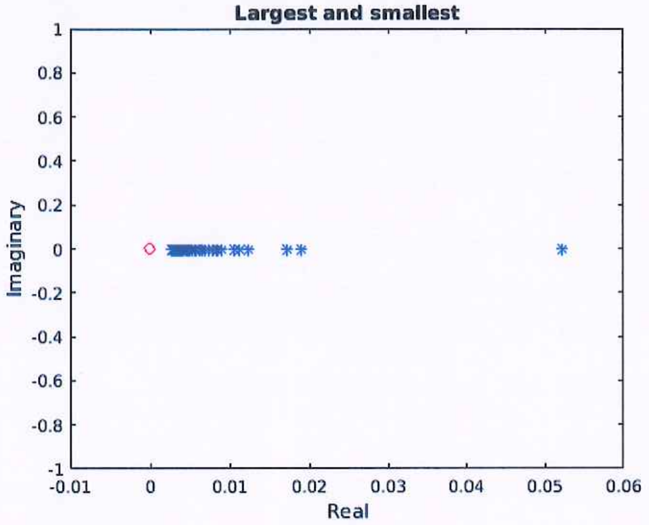
b) Linear system (K-I)x=b. Is this easier or more difficult to solve than (a). Why?

(Can be solved with standard eigenvalue bounds / theory.)

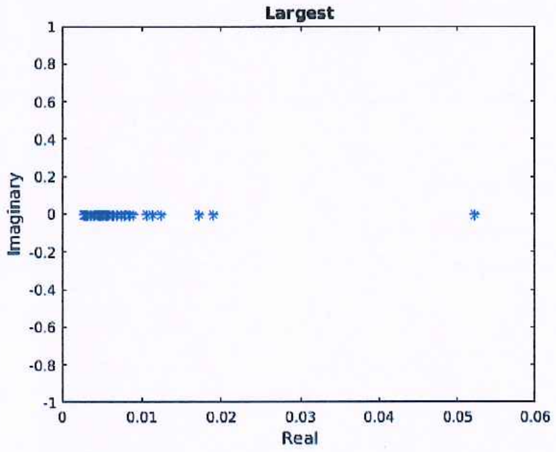
Solution 1-11

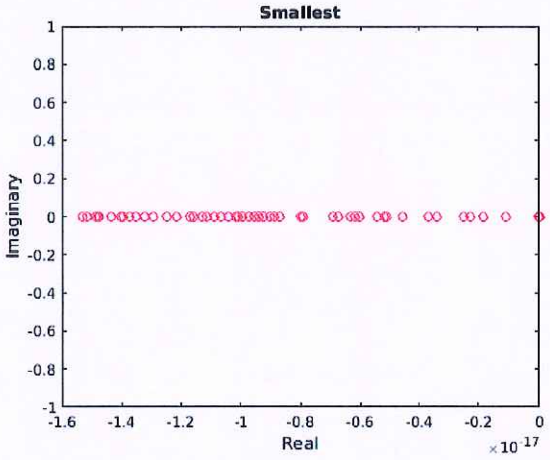
Solution by Emil:

a) The 50 largest and 50 smallest eigenvalues(measured in absolute value) of $\boldsymbol{K}$ were computed with EIGS in matlab, and a plotted in the figure below.



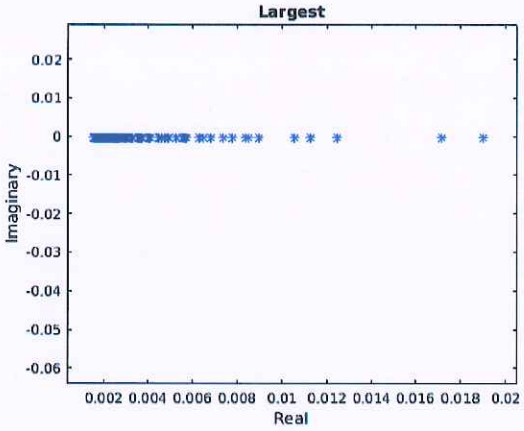
The blue dots are the largest, and the red circles are the smallest. To further magnify the difference we plot the separated in the two plots below.



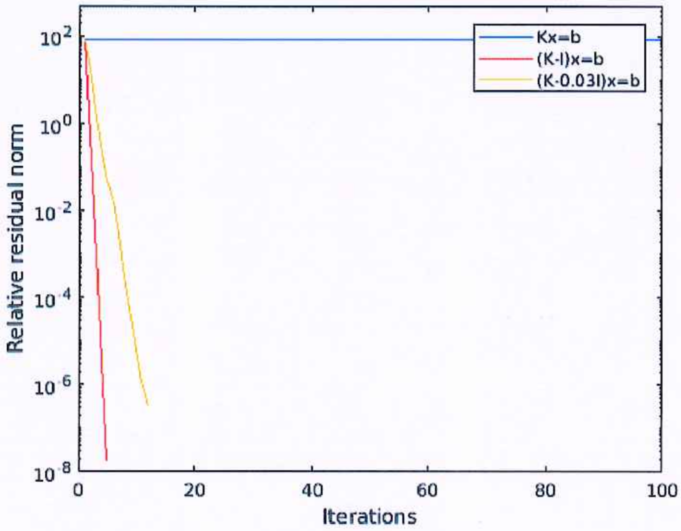


The eigenvalues are all spread out quite close to zero. For the explanation we assume that the the right-hand-side $\boldsymbol{b}$ does not have a "nice" structure, but rather is a linear combination of (almost) all eigenvectors. Hence for GMRES to converge, the Krylov space needs to contain a polynomial that efficiently approximates the matrix $\boldsymbol{K}$ on (almost) all eigenvalues. The spread of the eigenvalues makes a disc-reasoning impossible, and the approximation needs to be something along the lines of individually approximating each eigenvalue. Hence the polynomial needs to be of high degree, and hence many iterations are needed.

We also plot the 99 largest eigenvalues, excluding the largest. This further shows how bad the eigenvalue spread is.



b) The linear system $(K-I)x=b$ is much easier to solve. The matrix $(K-I)$ has the same eigenvalues as K, except that they are all shifted with -1. Thus given the eigenvalue spread above, we can conclude that the eigenvalues of $(K-I)$ will be spread very close to -1. With a disc-reasoning we then get that $\frac{\|r_m\|}{\|r_0\|} \leq \left(\frac{\rho}{c}\right)^m$ with $\rho \leq 0.06$ and $c=1$. Thus in at most 7 iterations the error is below 10^{-8}. We plot the convergence of a 100 iterations with GMRES for K, $(K-I)$, and $(K-0.03I)$. The last one also gives good convergence, since the eigenvalues are still clustered, but not too much around zero.

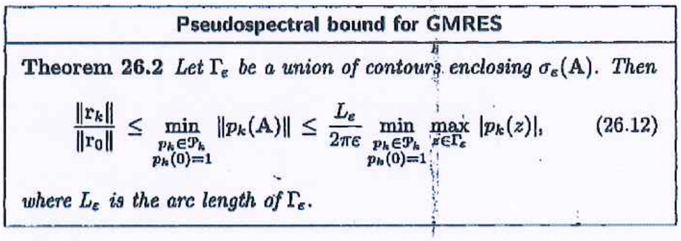


We see the poor convergence for K. But the shifted versions have good convergence (although there exists some shifts, like for example$(K-0.01I)$ that gives worse, but perhaps acceptable, convergence).

Moderator comment: Correct!

Problem 1-5

The pseudospectral bound in Trefethen&Embree:



Pseudospectral bound for GMRES

Theorem 26.2 *Let Γ_ε be a union of contours enclosing $\sigma_\varepsilon(\mathbf{A})$. Then*

$$\frac{\|\mathbf{r}_k\|}{\|\mathbf{r}_0\|} \le \min_{\substack{p_k \in \mathcal{P}_k \\ p_k(0)=1}} \|p_k(\mathbf{A})\| \le \frac{L_\varepsilon}{2\pi\varepsilon} \min_{\substack{p_k \in \mathcal{P}_k \\ p_k(0)=1}} \max_{z \in \Gamma_\varepsilon} |p_k(z)|, \qquad (26.12)$$

where L_ε is the arc length of Γ_ε.

A matrix has eigenvalues located at

10,11,12,13,....100

The matrix is normal, and it turns out that the epsilon-pseudospectra consists of a union of discs of radius epsilon (for all epsilon).

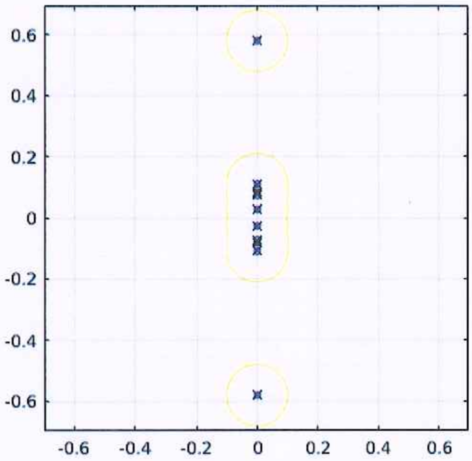
a) Derive a general formula containing epsilon (hint, reasoning similar to "Numerical linear algebra" course).

b) Visualize the bound for several different epsilon in a semilogy-plot. Which epsilon-value is most predictive for many iterations?

Solution 1-5

Problem 1-6

A matrix has the following pseudospectra for some value of ε:



Which of the matrices does it correspond to?

a) A=(tril(ones(10,10),1)-triu(ones(10,10),-1))/10;

b) A=(tril(ones(10,10),1)+triu(ones(10,10),-1))/10;

c) A=(tril(ones(10,10))+triu(ones(10,10)))/10;

d) A=(tril(ones(10,10))-triu(ones(10,10)))/10;

Justify your answer by identifying properties in the figure and properties of the matrix.

Not needed to solve the problem: The figure was generated by running

```
>> pscont(A,4,500);
```

The pscont.m command is available here:

https://se.mathworks.com/matlabcentral/fileexchange/2360-the-matrix-computation-toolbox?focused=5041522&tab=function

you also need cpltaxes.m.

Solution 1-6

Solution by Federico:

From the figure, we can see that all the eigenvalues are imaginary, and are pairs of complex conjugate. Thanks to this, by noticing that matrices *b)* and *c)* are symmetric and consequently have all real eigenvalues, we can rule them out.

Now the choice is between *a)* and *d)*. To see whether it is one or the other, we can check the largest eigenvalue in module. We check the one of matrix *d)*, using the power method. Now, because the matrix has two eigenvalues of largest module, but complex conjugate, λ_1 and λ_1^*, the power method would not converge.

So, instead of applying the power iteration to the matrix A corresponding to *d)*, we can apply it to A^2, which has two equal eigenvalues of same module, equal to the square of the module of the eigenvalues of A.

By running the power iteration, which will converge even if there are two identical eigenvalues with the biggest module (as seen in wiki exercise A16 from last year's NLA), we find the value $\mathbf{0.39863}$, which has square root $\mathbf{0.63137}$. This value is larger than the one in the figure, which lies just below $\mathbf{0.6}$, hence this must not be the matrix with the spectrum and pseudospectrum in the figure.

The answer is consequently *a)*.

Correct

Problem 1-17

Discovering properties of the pseudospectra:

(a) Prove that$\sigma_{|c|\varepsilon}(cA) = c\sigma_\varepsilon(A)$ where c is a complex number.

(b) Let D_ε be the disk (in the complex plane) centered in zero with radius ε. For which class of matrices it holds $\sigma_\varepsilon(A) = \sigma(A) + D_\varepsilon$?

Hint: start testing this properties for diagonal matrices, Jordan blocks, 2x2 matrices, etc. No closed answer required.

If it is too hard, just show that $\sigma_\varepsilon(A) \supseteq \sigma(A) + D_\varepsilon$. (Emil: I flipped the inclusion to match Theorem 2.2 in Trefethen and Embree)

(c) It is known that if A and B are diagonalizable and commute, i.e., $AB = BA$, then $\sigma(A) + \sigma(B) = \sigma(A + B)$. Is this true for the pseudospectra? More precisely is $\sigma_\varepsilon(A) + \sigma_\varepsilon(B) = \sigma_\varepsilon(A + B)$? If yes prove it, otherwise show a counterexample.

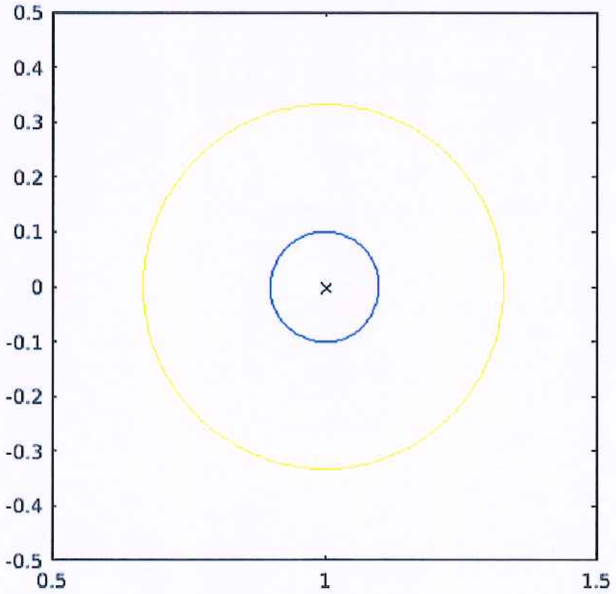
Solution 1-17

Solution by Emil:

a) We interpret $c\sigma_\epsilon(A) = \{z = cx \mid x \in \sigma_\epsilon(A)\}$, which in turn means (by definition) that x is any complex number such that $\|(x - A)^{-1}\| > \epsilon^{-1}$. This is equivalent to that z is any number such that $\|(\frac{z}{c} - A)^{-1}\| > \epsilon^{-1}$, which can be further simplified to $\|(z - cA)^{-1}\| > (|c|\epsilon)^{-1}$. Thus we conclude that $z \in \sigma_{|c|\epsilon}(cA)$. Finally we know that for any numbers $x \in \sigma_\epsilon(A)$ and $z \in \sigma_{|c|\epsilon}(cA)$ we have the relation $z = cx$, which shows that $c\sigma_\epsilon(A) = \sigma_{|c|\epsilon}(cA)$.

b)Trefethen and Embree, Theorem 2.2 states that if the norm is the 2-norm, then the equality is true if and only if A is normal.

The equality obviously fails for the simplest type of Jordan blocks of size 2x2. Consider $A = (1,1;0,1)$, which has a double eigenvalue in 1. Perturb it with $E = (0,0;\varepsilon,0)$. Then we can see that $\sigma(A+E) = \{1 \pm \sqrt{\varepsilon}\}$, which for an $\varepsilon < 1$ is outside the ball D_ε. The example is illustrated in the following plot, in which the pseudospectra is shown in 10-log scale for$\varepsilon = 10^{-1}, 10^{-2}$ (blue respective yellow circles).



Nice

```
pscont([1,1;0,1], 4, 100, [0.5, 1.5, -0.5, 0.5], [-2:-1])
```

The proof for the inclusion is $\sigma_\varepsilon(A) \supseteq \sigma(A) + D_\varepsilon$ is constructive in manner. Using the second definition of pseudo-spectra we observe that for the specific choice $E = \delta I$, with $\delta \in \mathbb{C}$ and $0 < |\delta| < \varepsilon$, we have that $\|E\| < \varepsilon$ and that $\sigma(A+E) = \sigma(A) + \delta$. Thus having δ arbitrary in the open ball D_ε gives the result.

c) This is not true. Consider the 2-norm, and A and B normal. Then by the above cited theorem $\sigma_\varepsilon(A) = \sigma(A) + D_\varepsilon$ and $\sigma_\varepsilon(B) = \sigma(B) + D_\varepsilon$. Thus $\sigma_\varepsilon(A) + \sigma_\varepsilon(B) = \sigma(A) + D_\varepsilon + \sigma(B) + D_\varepsilon = \sigma(A+B) + 2D_\varepsilon = \sigma(A+B) + D_{2\varepsilon}$ which is not equal to $\sigma_\varepsilon(A+B) = \sigma(A+B) + D_\varepsilon$.

In words one could say that when constructing $\sigma_\varepsilon(A)$ and $\sigma_\varepsilon(B)$ separately, one can do a maximal and constructive perturbation in the eigenvalue corresponding to the same eigenvector for both A and B and thus get a larger total perturbation in the sum, than one could do on the matrix $A + B$.

Moderator comment: Correct. (b) is difficult since the question sounds like sufficient and necessary conditions for pseudospectra to consist of disc. (c): great insightful proof!

Problem 1-18

Problem by Emil:

In this problem we look at eigenvalue sensitivity, pseudospectra, and GMRES convergence.

a) Consider the following MATLAB code: randn('seed',0)

```
[Q,~] = qr(rand(n,n));
a = [-30-3*rand(n/3,1); -20-2*rand(n/3,1); -10-2*rand(n/3,1)];
A = Q*diag(a)*Q';
A = (A+A')/2;

S = balance(compan(poly(a)));
```

It creates a "well-behaved" matrix A which is symmetric and with prescribed eigenvalues in three regions of the complex plane (Real line). The matrix S is a balanced version of the companion matrix to A, and hence mathematically it has the same eigenvalues as A. Thus from an eigenvalue analysis, GMRES should have similar convergence behavior. However, the operations involved in constructing S are not well conditioned.

How can we see this if we compute the eigenvalues and the pseudospectra? (try n = 12 and 21, a plot like pscont(A, 4, 100, [-40,0,-4,4]) could be nice).

Nice Q

b) Now instead consider the MATLAB code:

```
[Q,~] = qr(rand(n,n));
a = [-30-3*rand(n/3,1); -20-2*rand(n/3,1); -10-2*rand(n/3,1)];
S = balance(compan(poly(a)));
aa = eig(S);
A = Q*diag(aa)*Q';
```

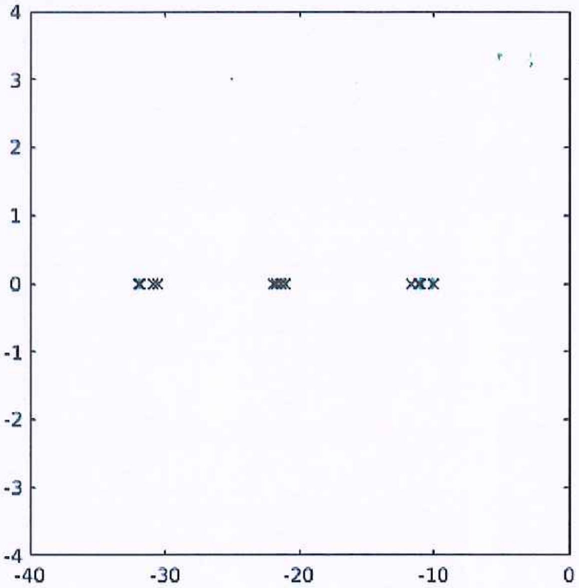
It works similarly as above, but here we are more sure that A and S have the same eigenvalues, although not fully as nicely distributed. create a random right-hand-side and consider the GMRES convergence behavior.

What can we say? How does the GMRES convergence look like? How does the pseudospectra of A and S look like? (this is slightly more stable and hence we can consider n = 66, a plot like pscont(A, 4, 100, [-80,0,-45,45], -10:1:0) could be nice).

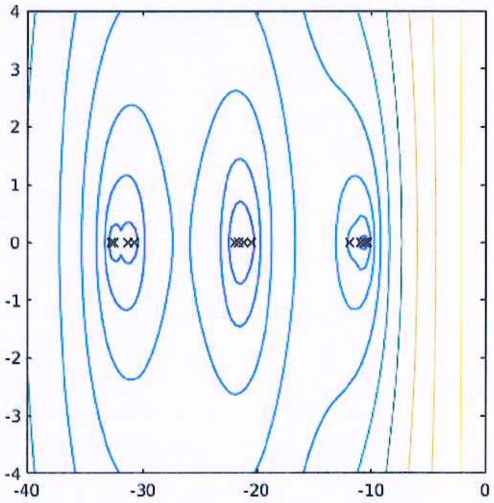
Solution 1-18

Before discussing the solution, I point out that this exercise contains a very important message: the pseudospectra depends on the eigenvectors. Two matrices may have the same eigenvalues but different pseudospectra and in general, the sensitivity of the eigenvalues is described in function of the eigenvectors, see this link or the Gene Golub book (Matrix computation). Therefore, it is not enough to know the spectrum of a matrix in order to fully describe the behavior of GMRES.

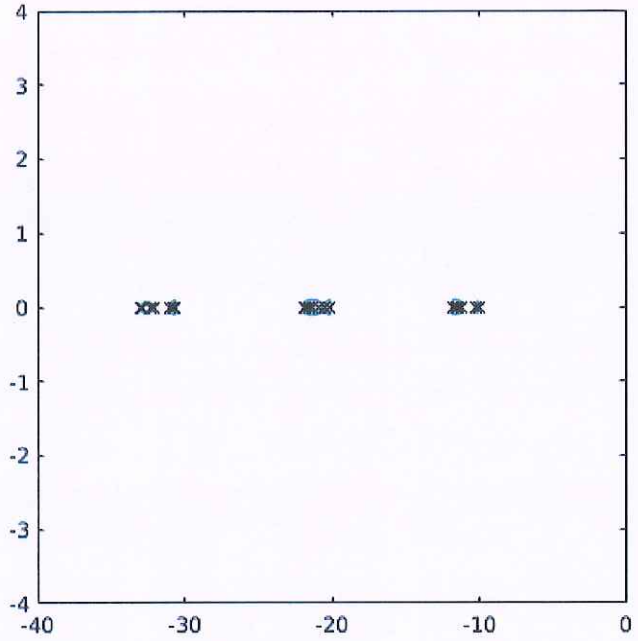
(a) For n=12 the following is the pseudospectra of A



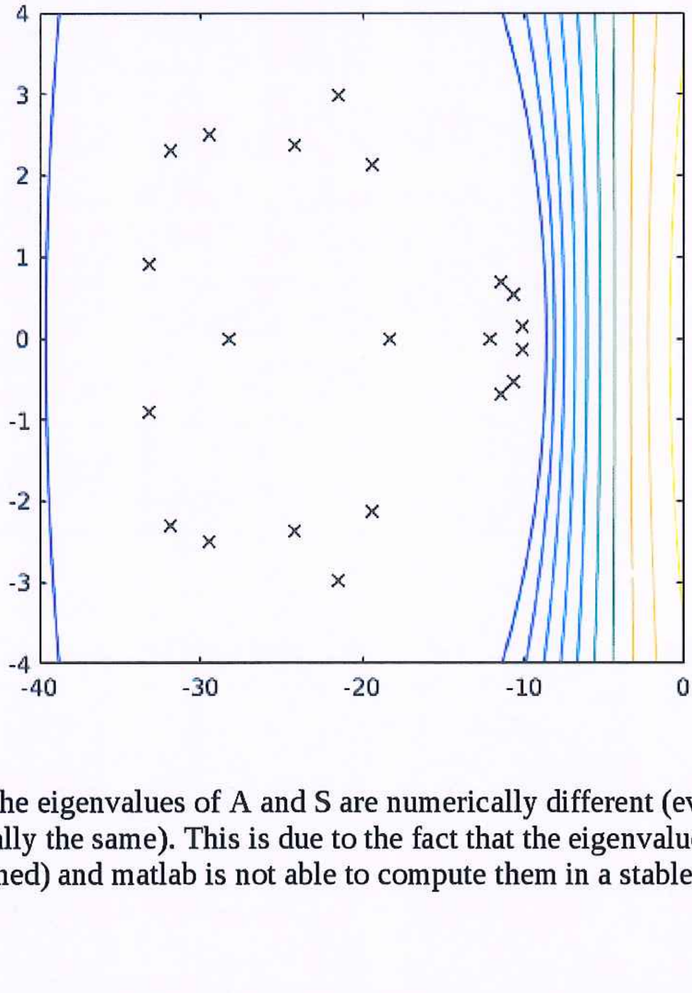
whereas the following is the pseudospectra of S



For n=21 the following is the pseudospectra of A



whereas the following is the pseudospectra of S

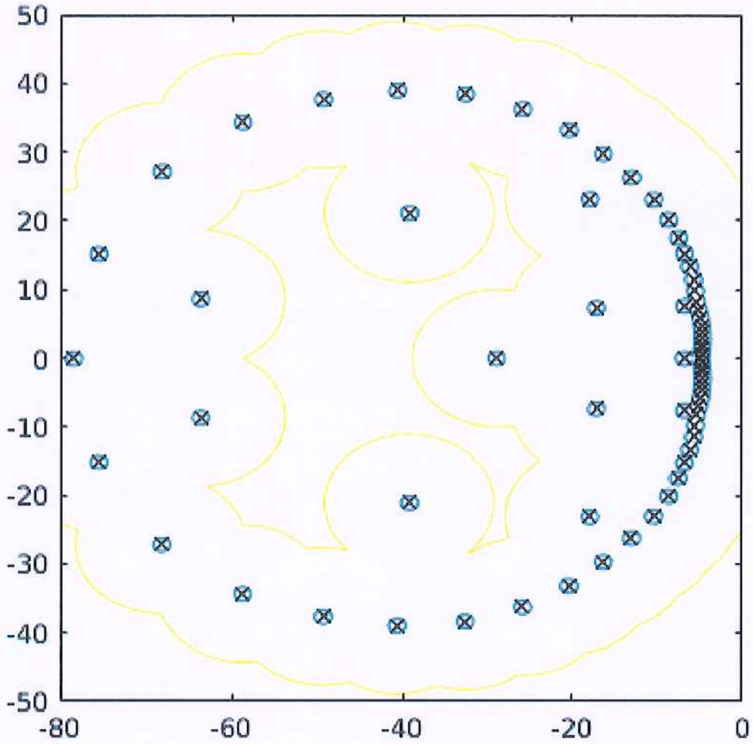


Notice that the eigenvalues of A and S are numerically different (even if by construction they are mathematically the same). This is due to the fact that the eigenvalues of S are too sensitive to small changes (ill-conditioned) and matlab is not able to compute them in a stable way.

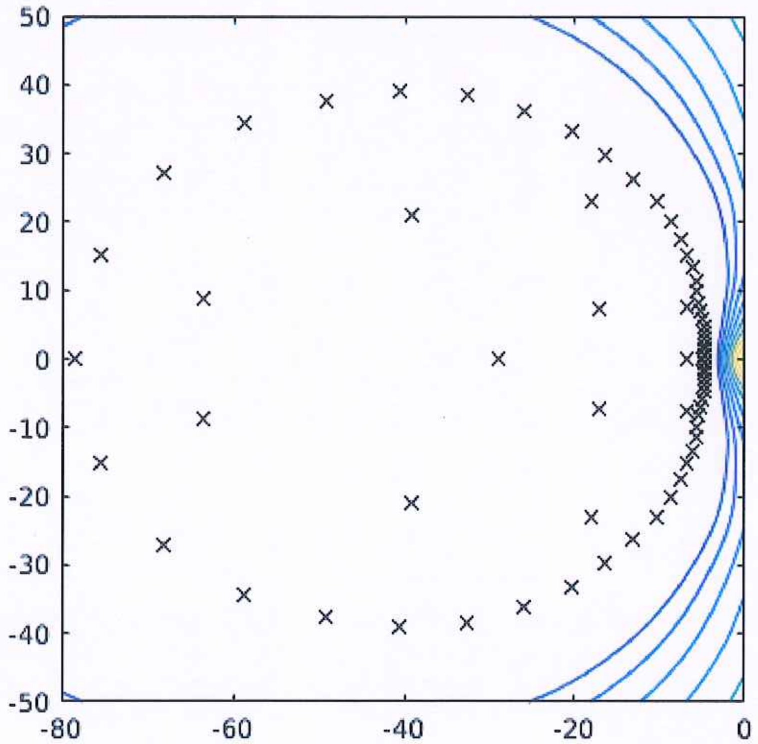
(b)

Even if the matrix S is now constructed in a more stable way, all the reasoning of the point (a) is still valid. We now consider n=66.

The following is the pseudospectra of A (generated as pscont(A, 4, 500, [-80,0,-50,50],-10:1:1);)

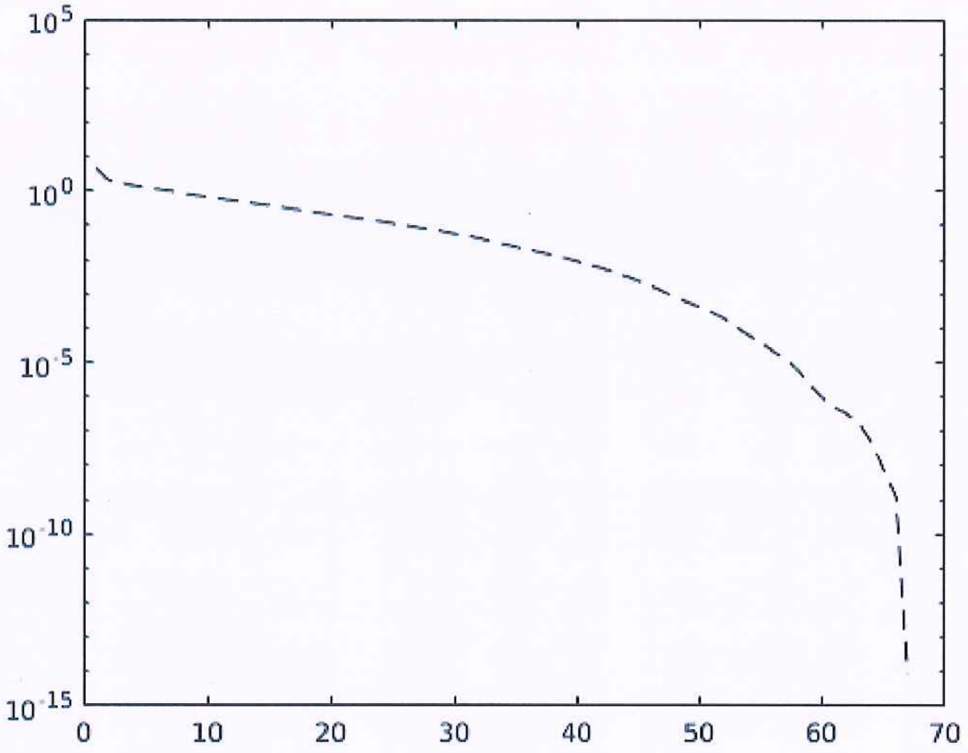


whereas the following is the pseudospectra of S (generated as pscont(S, 4, 500, [-80,0,-50,50],-10:1:1);).

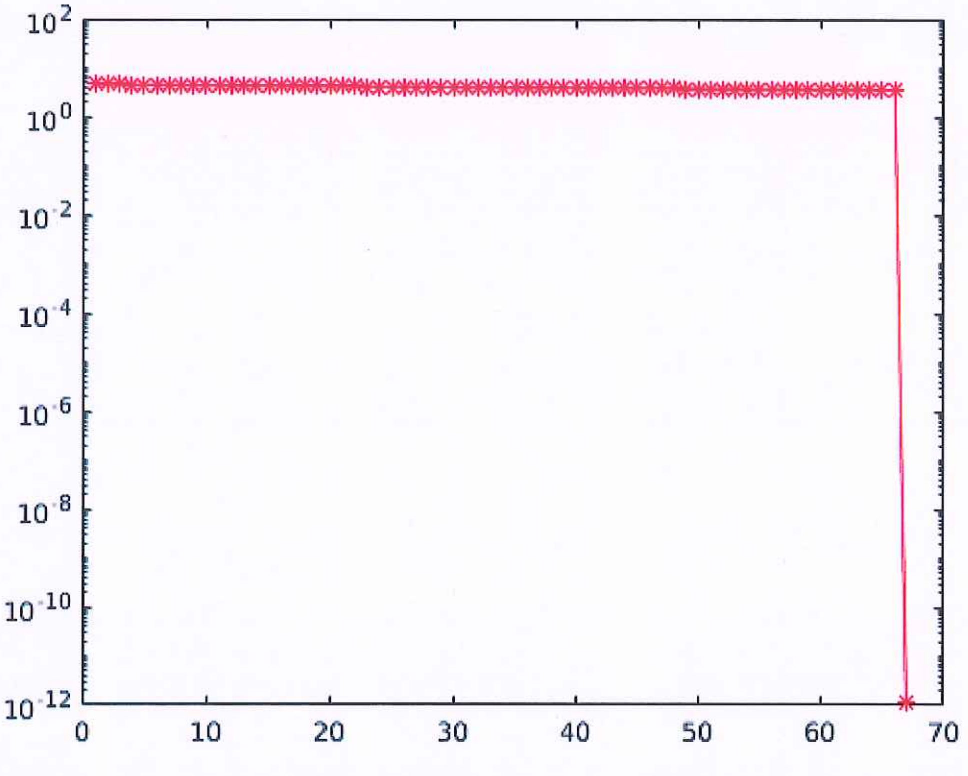


Nice

Here we can see the convergence of gmres for A



whereas here we see the convergence of gmres for S



Clearly for S gmres does not converge even if A and S have the same eigenvalues. This is explained by the reasoning in (a). The convergence of GMRES does not depend only on the eigenvalues but also on their sensitivity (described by the eigenvectors). By using the theory that we saw in the class we can use the pseudospectra bound for the convergence of gmres we have

$$\frac{\|r_m\|}{\|r_0\|} \leq \frac{L_\varepsilon}{2\pi\varepsilon} \min_{\substack{deg(p)\leq m\\ p(0)=1}} \max_{\Gamma_\varepsilon} |p(z)|$$

We can clearly see that for the matrix S the constant L_ε is very large, therefore the transient phase will be very long.

A short (and naive) answer to this second part can be given by using the standard theory for the convergence of GMRES. More precisely we have

$$\frac{\|r_m\|}{\|r_0\|} \leq \kappa(V_A) \min_{\substack{deg(p)\leq m\\ p(0)=1}} \max_{\lambda\in\sigma(A)} |p(z)|$$

FIXED: Moderator comment: \cond(A) should be \cond(V), where V eigenvector matrix. See GMRES on wikipedia

Where V_A is the matrix containing as columns the eigenvectors of A. We now compute the condition numbers $\kappa(V_A) = 1$ and $\kappa(V_S) = 1.7572e + 16$

Conclusion:
for gmres we have that the asymptotic convergence is characterized by the spectrum whereas the transient phase is depends on the sensitivity of the eigenvalue (that is related to the eigenvectors) and it is characterized by the pseudospectra.

Moderator comment: Great solution. I agree that "the pseudospectra depends on the eigenvectors". I would phrase it as "the pseudospectra of diagonalizable matrices can be described with eigenvectors".

Problem 1-19

Problem by Emil:

This problem relates to 1-17 b), but you do not need to solve that one to do this.

a) Let $\sigma_\varepsilon(A)$ be the 2-norm pseudospectrum, let $W(A)$ be the field of values (also called numerical range), i.e., $W(A) = \{z \,|\, z = v^*Av,\ v \in \mathbb{C}^n,\ \|v\|_2 = 1\}$, and let D_ε be a ball of radius ε centered at the origin.

Show that $\sigma_\varepsilon(A) \subseteq W(A) + D_\varepsilon$.

b) Show that for $\varepsilon, \delta > 0$, then $\sigma_\varepsilon(A) + D_\delta \subseteq \sigma_{\varepsilon+\delta}(A)$. Here σ_ε can be any norm-pseudospectra.

Solution 1-19

Solution by Federico:

a) From the third definition of the pseudospectra (Higham and Embree, *Spectra and Pseudospectra*, p.16):

$$\sigma_\varepsilon(A) = \{z \in \mathbb{C} \text{ s.t. } \exists\, v \in \mathbb{C},\ \|v\| = 1,\ \|(zI - A)v\| < \varepsilon\}$$

and we want to prove that, for the 2-norm pseudospectrum, $z = y + x$, where $y \in W(A)$ and $\|x\|_2 < \varepsilon$.

Let $z \in \sigma_\varepsilon(A)$, and v the vector from the definition; let $w = v^*(zI - A)v$. Then

$$w = zv^*v - v^*Av = z\|v\|_2^2 - v^*Av = z - v^*Av$$

hence $z = w + v^*Av = w + u$, where $\|w\|_2 \leq \|v^*\|_2\|(zI - A)v\|_2 < \varepsilon$ and $u \in W(A)$. The thesis is proved.

b) Let $z \in \sigma_\varepsilon(A)$, and let v unitary such that $\|(zI - A)v\| < \varepsilon$; let $x \in D_\delta$. Then

$$\|((z + x)I - A)v\| = \|(zI - A)v + xv\| \leq \|(zI - A)v\| + \|x\|\|v\| < \varepsilon + \delta$$

Nice

which proves that $z + x \in \sigma_{\varepsilon+\delta}(A)$.

Problem 1-20

Problem by Emil: (inspired by GVL section 7.9)

For powers of matrices we have that if the spectral radius is less than 1, $\rho(A) < 1$, then $\lim_{k \leftarrow \infty} \|A^k\| = 0$. From pseudospectra we also have the following bound: $\sup_{k \geq 0} \|A^k\|_2 \geq \frac{\rho_\varepsilon(A)-1)}{\varepsilon}$, where $\rho_\varepsilon(A)$ is the pseudospectra spectral radius, i.e., $\rho_\varepsilon(A) = \max_{\lambda \in \sigma_\varepsilon(A)} |\lambda|$.

a) Consider the matrix

```
A = [0.99, 100; 0, 0.99]
```

Plot $\|A^k\|$ for many different values of k (like $k = 1, 2, \ldots, 2000$). What can we see? Does it fit the bound? Can you find an ε that gives an approximation of the "bump"?

b) Is the phenomena due to numerical computations? Try with making A symbolic. When is this an issue in practice?

c) Prove the bound. (This is technical, and hence a few hints/outline is provided)

Hints:

1) If $\|A^k\| \leq C$ for all $k = 0, 1, \ldots$, then $(A - zI)^{-1} = \sum_{k=0}^{\infty} \frac{-1}{z^{k+1}} A^k$ for all $|z| > 1$.

2) Consider $\|(A - zI)^{-1}\| = \varepsilon^{-1}$, which we write as $\|(A - zI)^{-1}\| = \frac{K}{|z|-1}$ for $K = \frac{\rho_\varepsilon(A)-1}{\varepsilon}$ and $|z| = \rho_\varepsilon(A)$, where we assume that $\rho_\varepsilon(A) > 1$.

3) Prove that $K \leq \sup_{k \geq 0} \|A^k\|$ by observing that in the case of the rewriting in 2) then by using 1), one can show that $\frac{K}{|z|-1|} \leq \frac{1}{|z|}\left(1 + \frac{\sup_{k \geq 0} \|A^k\|}{|z|-1}\right)$.

4) Observe that for pseudospectrum $\|(A - zI)^{-1}\| > \varepsilon^{-1}$, but this can be written as $\|(A - zI)^{-1}\| = \varepsilon^{-1} + \delta$, so that the relation in 3) is still valid.

Solution 1-20

Problem 1-35

a) Derive an explicit expression for the pseudospectra of a diagonal matrix A

b) Based on the explicit expression in a derive a family of bounds for GMRES (independent of RHS-vector b)

c) Provide a plot which compares the bound for different eps-values and the residual norm. You can use matlab GMRES-implementation explained in Problem 1 15.

Solution 1-35

Problem 1-36

Solution 1-36

RHS-dependent bounds

Problem 1-10

Krylov subspaces properties and RHS

We want to use GMRES for solving $Ax = b$ where $A \in \mathbb{R}^{n \times n}$

(a) Assume that b is a linear combination of $m \ll n$ eigenvectors of A (for example $m = 10$ eigenvectors). After exactly how many iterations GMRES converges?

(Elias: You can assume the linear combination has non-zero coeff's, i.e., $b = a_1 x_1 + \cdots a_m x_m$ where a_i are non-zero)

(b) Let W an invariant space for A, i.e., $AW \subseteq W$ with $dim(W) = m \ll n$. Assume that $b \in W$. After exactly how many iterations GMRES converges?

Hint: these properties directly come from the Krylov space structure. Try to answer the question for $m = 1$ and then generalize.

Solution 1-10

Solution by Parikshit:

(a) Since we are given that $A \in \mathbb{R}^{nxn}$ has a full set of n eigenvectors, it is diagonalizable as $A = Z\Lambda Z^{-1}$.

Problem author comments: I did not write that the matrix A is diagonalizable and not all the matrices are diagonalizable.

Also, since b can be written as a linear combination of exactly m eigenvectors, we can write

$$b = Zw, \text{ where } w_i = \begin{cases} w_i \neq 0, & 1 \leq i \leq m \\ 0, & m+1 \leq i \leq n \end{cases}$$

Note that we have assumed without loss of generality that the m eigenvectors we refer to occur as the first m columns of Z

Hence, using the RHS bound result for the GMRES residual and substituting for w, we have,

$$\frac{||r_m||}{||r_0||} \leq ||Z|| min_{p \in P_m^0} \left(\sum_{i=1}^{m} |w_i|^2 |p(\lambda_i)|^2 \right)^{1/2}$$

From the above inequality we can make the upper bound zero by choosing p to be a polynomial with atmost m distinct roots/eigenvalues $\lambda_i, \quad i = 1, \ldots, m$, which we are allowed to do since

$p \in P_m^0$. Hence, we have $0 \leq ||r_m||/||r_0|| \leq 0 \implies r_m = 0$. This means that GMRES will converge after a maximum of exactly $m = 10$ iterations. The convergence might occur before 10 iterations if all the eigenvalues are not unique.

Problem author comments: even if you added the hypothesis that that matrix is diagonalizable, I accept this solution that is correct with this extra hypothesis. However you will not be able to do the part (b) of this exercise with the same argument. Tell me if you want an extra hint.

Alt. Sol.
good!

Moderator comment: Correct (under additional assumption).

Problem 1-25

Solution 1-25

Problem 1-26

Learning RHS-bounds on your own. This is suitable to be solved before careful reading of the RHS-paper.

In the numerical linear algebra course we learned that GMRES generated iterates satisfying:

$\|r_m\| = \min_{p \in P_m^0} \|p(A)b\|$

Assume that $A = Z\Lambda Z^{-1}$ is a diagonalization (Λ is a diagonal matrix). Let $w = Z^{-1}b/\|b\|$.

a) Show that $p(A)b = ZWp(\Lambda)e$. What is W? what is e?

b) Show that $\|r_m\| \leq \|Z\| \min_{p \in P_m^0} \|Wp(\Lambda)e\|_2$. What are the elements of $Wp(\Lambda)e$ explicitly?

c) How is the derivation related to the RHS-paper? (It's not exactly the same.)

Solution 1-26

Solution by Parikshit:

(a) We are given that A is diagonalizable as $A = Z\Lambda Z^{-1}$ and $Zw = b/||b||$

Hence, substituting for A and b, we get,

$p(A)b = p(Z\Lambda Z^{-1})Zw||b|| = Zp(\Lambda)w||b||$ Equation(1)

Letting $W = ||b||diag(w) \implies w||b|| = We$, where e is a vector of all ones.

This leads to

$Zp(\Lambda)w||b|| = Zp(\Lambda)We = ZWp(\Lambda)e$ Equation(2)

(Since multiplication of diagonal matrices is commutative)

Good

Combining Equation(1) and Equation(2) completes the proof.

(b) Using the result from the numerical linear algebra course and substituting for $p(A)b$

using the result from (a), we have,

$||r_m|| \leq min_{p\in P_m^0}||ZWp(\Lambda)e|| \leq ||Z||min_{p\in P_m^0}||Wp(\Lambda)e||$

Note that $Wp(\Lambda)e$ is a vector whose i-th element is $||b||w_i p(\lambda_i)$, where w_i is the coefficient of the i-th eigenvector when $b/||b||$ is written as a linear combination of the eigenvectors(i.e. columns in Z)

(c) The derivation in the paper is different in the following way:

Instead of assuming $ZW = b/||b||$, the assumption is $ZW = r_0/||r_0||$, where r_0 is the initial residual.(Note that $r_0 = b$ if $x_0 = 0$)

Moderator comment: Great solution!

Problem 1-14

RHS-bounds for Jordan blocks. Jordan blocks are not as scary as they appear.

a) Adapt the RHS-bound result (in the paper or Problem 1 26) to a matrix with a Jordan block. That is, show a corresponding bound is when the matrix is not diagonalizable but

$$A = Z \begin{bmatrix} J & & & \\ & \lambda_k & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} Z^{-1}$$

where J is a k x k Jordan block.

b) Derive an explicit formula for p(J) when p is a polynomial, and find a polynomial p\in P_m^0 such that p(J) is a diagonal matrix and p(z)=g(z)h(z) where g\in P_{m-k+1}.

c) Specialize the bound in (a) by using (b) and derive a result which directly depends on the eigenvalue location and weights.

Solution 1-14

Problem 1-8

Setup a sparse linear system of equations using FEM-solver system FeNICS. You will need to install the python framework for Fenics. This tutorial may be of use

https://fenicsproject.org/pub/tutorial/sphinx1/._ftut1004.html

Moderator comment: This problem can help us understand the performance of the preconditioners we use later. So downloadable easy-to-use mat-files would be nice.

Solution 1-8

The solution is very technical and I think I did not understand every detail. However I succeeded to generate sparse matrices form fenics and import them into matlab.

I will now point what you need to do and then I will attach three examples in fenics that you can reverse-engineering and generalize to other PDEs.

PART 1 (python-fenics):

You need to add in the python script the following lines (in the beginning):

- parameters['reorder_dofs_serial'] = False # to avoid reordering/permutations of the matrices
- from scipy.io import savemat # to save the matrices
- parameters.linear_algebra_backend = "Eigen" # too technical to explain

After this, you can use fenics to define the PDE. Then you have to construct the stiffness matrix, convert it to CSR data, extract the sparse array format and save it in a format that matlab can read. In compact, this is done with the following lines of code that you have to add at the end of the script

- A, b = assemble_system(a, L, bc);
  rows,cols,vals = as_backend_type(A).data() # Get CSR data
  A = as_backend_type(A).sparray()
  savemat('Elasticity.mat', {'A': A,'b':b.array()})

Well done!

PART 2 (examples):

The .py files are scripts in python that generate a .mat file that you can load with matlab. The .mat file contains matrix and RHS of the linear systems derived by a FEM discretization of the associated PDE.

- Example 1: a standard Poisson problem poisson_membrane.py
- Example 2: a Poisson problem with a more challenging domain (square with an hole in the middle) poisson_membrane_2.py
- Example 3: a 2d elasticity problem elesticity.py
- If you do not have Fenics yet and you just want to see/test the matrices, just load the files Poisson.mat, Poisson_2.mat, Elasticity.mat

---

Problem 1-15

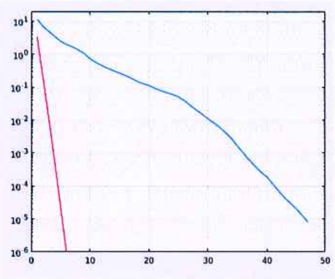
Run GMRES on a diagonal matrix with consisting of elements 1:n

```
>> n=100;
>> A=spdiags((1:100)',0,100,100);
>> b=randn(100,1);
```

The matlab implementation of GMRES (without restarting) can be run in this way in order to get a convergence diagram:

```
>> [X,FLAG,RELRES,ITER,RESVEC] = gmres(A,b,length(b));
>> semilogy(RESVEC)
```

You get the following convergence diagram



One of the curves is with random RHS b one and the other with

```
>> b=zeros(n,1); b(50:60)=1;
```

(a) Which RHS correspond to which curve? Relate to theory for RHS-convergence of GMRES

(b) If we take RHS: b=zeros(n,1); b(90:100)=1; Do we get faster or slower convergence. Relate to RHS-convergence theory

Solution 1-15

Solution by Emil:

a) The fast (red) curve relates to the structured RHS (b=zeros(n,1); b(50:60)=1;) and the slow (blue curve) relates to the random RHS.

The matrix A is a diagonalizable matrix (it is diagonal and hence trivially diagonalizable). The eigenvectors are the corresponding unit vectors e_i. By considering Theorem 2.2 in Titley-Peloquin, Pestana, and Wathen, the GMRES bound can be written as $\frac{\|r_m\|}{\|r_0\|} \leq \min_{\substack{q\in\Pi_m \\ q(0)=1}} \left(\sum_{i=1}^{n} |w_i|^2 |q(\lambda_i)|^2\right)^{1/2}$, where Π_m is the set of polynomials of degree max m, and w_i is the the projection coefficient of b (the RHS) onto the i-th basis vector in the eigenvector basis. For this problem, that means that w_i is simply the i-th component in the b, i.e., $w_i = b_i$. With that in mind, we can, for the structured b, rewrite the bound as $\frac{\|r_m\|}{\|r_0\|} \leq \min_{\substack{q\in\Pi_m \\ q(0)=1}} \left(\sum_{i=50}^{60} |q(\lambda_i)|^2\right)^{1/2}$. Thus full convergence is guaranteed in only 11 iterations. However, with a disc-reasoning based on this bound we can say that $\frac{\|r_m\|}{\|r_0\|} \leq \left(\frac{\rho}{c}\right)^m$, where we can now take $\rho = 5$, and $c = 55$, since the only eigenvalues involved in the last sum are $50, 51, \ldots, 60$. Thus an error of 10^{-6} is reached in at most 6 iterations, which is exactly what is observed in the figure.

This type of reasoning is not possible for a random b since in general we will have $w_i \neq 0$ for all i. Some eigenvalues can be "down prioritized", but that is still not giving the same acceleration and simple bounds to compute. A simplified bound could be argued for by using 10 iterations to "remove" the eigenvalues 1-10. Then a disc-argument on the rest would give something like $\rho = 40$, and $c = 50$, which would give that after 10+40 = 50 iterations the error would be bounded by $1.33 \cdot 10^{-4}$. This is simplified behavior is not exactly what is observed, but it goes somewhat along with the observed convergence.

Nice

b) If the RHS is changed (to b=zeros(n,1); b(90:100)=1;) the convergence will be faster. One can do a similar argument as the one in a), both in terms of only regarding some eigenvalues, and also the disc-reasoning. However, this time the disc would be $\rho = 5$, and $c = 95$ and thus providing an error of 10^{-6} in at most 5 iterations. Compare the convergence rates $5/95 \approx 0.053$ and $5/55 \approx 0.091$.

Moderator comment: Correct.

Problem 1-27

Problem by Parikshit:

Prove that if B can be diagonalized as $B = Z\Lambda Z^{-1}$, then for any arbitrary non-singular matrix $K \in \mathbb{C}^{n\times n}$, the GMRES residuals satisfy the following inequality,

$$\frac{||r_k||}{||r_0||} \leq ||K||_2 ||K^{-1}\tilde{r}||_2 \kappa_2(K^{-1}Z) min_{q\in P_0^k} max_{\lambda\in\sigma(B)} |q(\lambda)|$$

where $\tilde{r} = \frac{r_0}{||r_0||_2}$

Solution 1-27

Solution by Federico:

To prove this, we start from the original form for the norm of the residual:

$$\|r_k\|_2 = \min_{q \in P_0^k} \|q(B) r_0\|_2$$

where P_0^k is the space of polynomials of degree at most k such that have value 1 in 0. We start by bringing the norm of r_0 inside the norm on the RHS, and use the identity $I = K^{-1}K$:

$$\frac{\|r_k\|_2}{\|r_0\|_2} = \min_{q \in P_0^k} \left\| q(B) \frac{r_0}{\|r_0\|_2} \right\|_2 = \min_{q \in P_0^k} \left\| K K^{-1} q(B) K K^{-1} \frac{r_0}{\|r_0\|_2} \right\|_2 \leq$$

Nice

$$\leq \|K\|_2 \|K^{-1} \tilde{r}\|_2 \min_{q \in P_0^k} \left\| K^{-1} q(Z \Lambda Z^{-1}) K \right\|_2 =$$

because of the presence of Z and its inverse, the polynomial can be written as $Zq(B)Z^{-1}$:

$$\leq \|K\|_2 \|K^{-1}\tilde{r}\|_2 \|K^{-1}Z\|_2 \|KZ^{-1}\|_2 \min_{q \in P_0^k} \|q(\Lambda)\|_2 = \|K\|_2 \|K^{-1}\tilde{r}\|_2 \kappa_2(K^{-1}Z) \min_{q \in P_0^k} \|q(\Lambda)\|_2 \leq$$

$$\leq \|K\|_2 \|K^{-1}\tilde{r}\|_2 \kappa_2(K^{-1}Z) \min_{q \in P_0^k} \max_{\lambda \in \sigma(B)} |q(\lambda)|$$

and the lemma is proved.

Problem 1-28

Solution 1-28

Problem 1-29

Solution 1-29

Problem 1-30

Solution 1-30

Problem 1-40

Solution 1-40

Problem 1-41

Solution 1-41

Incorporating preconditioners in iterative methods

Left-right-split preconditioners. Flexible GMRES.

Problem 1-13

What is

a) Left preconditioning?

b) Right preconditioning?

c) Split preconditioning?

Solution 1-13

Assume we want to solve the system $\boldsymbol{Ax} = \boldsymbol{b}$. Furthermore let the matrix $\boldsymbol{M}$ be a preconditioner.

a) Left preconditioning means applying the preconditioner from the left

$$\boldsymbol{M}^{-1}\boldsymbol{Ax} = \boldsymbol{M}^{-1}\boldsymbol{b}.$$

b) Right preconditioning means applying the preconditioner from the right

$$\boldsymbol{AM}^{-1}\boldsymbol{u} = \boldsymbol{b}, \qquad \boldsymbol{x} = \boldsymbol{M}^{-1}\boldsymbol{u}.$$

Note that this corresponds to the change of variable $\boldsymbol{u} = \boldsymbol{Mx}$.

c) Assume $\boldsymbol{M}$ is factored $\boldsymbol{M} = \boldsymbol{M_L M_R}$. Typically $\boldsymbol{M_L}$ and $\boldsymbol{M_R}$ are triangular matrices. The preconditioning can then be split

$$\boldsymbol{M}_L^{-1}\boldsymbol{AM}_R^{-1}\boldsymbol{u} = \boldsymbol{M}_L^{-1}\boldsymbol{b}, \qquad \boldsymbol{x} = \boldsymbol{M}_R^{-1}\boldsymbol{u}.$$

Note that split preconditioning may be used to preserve symmetry, since we can select $\boldsymbol{M_L}$ and $\boldsymbol{M_R}$ such that $\boldsymbol{M}_L^{-1}\boldsymbol{AM}_R^{-1}$ is symmetric if A is symmetric.

Moderator comment: Correct!

Problem 1-21

Flexible GMRES with BICGSTAB as preconditioner

We want to solve (in matlab) the system $Ax = b$ with $A \in \mathbb{R}^{n \times n}$ with flexible GMRES and BICGSTAB (with variable number of iterations) as preconditioner.

(a) Complete the following script of flexible GMRES. More precisely as preconditioner $M_j x$ we consider the function M(j,x) that perform j steps of bicgstabl (implemented in matlab).

```
n=200;
e = ones(n,1);
A = spdiags([e 2*e e], -1:1, n, n);

b=rand(n,1);
norm_b=norm(b);

V(:,1)=b/norm_b;

m=30;
M=@(j,b) .......................... ;

for j=1:m
    Z(:,j)=M(j,................);
    V(:,j+1)=...............;
    h=V(:,1:j)'*V(:,j+1);
    V(:,j+1)=V(:,j+1)-V(:,1:j)*h;
    g=V(:,1:j)'*V(:,j+1);
    V(:,j+1)=V(:,j+1)-V(:,1:j)*g;
    H(1:j,j)=h+g;
    H(j+1,j)=norm(V(:,j+1));
    V(:,j+1)=V(:,j+1)/H(j+1,j);

    e1=eye(j+1); e1=e1(:,1);
    y=H\(norm_b*e1);
    xx=.........;
    err(j)=norm(A*xx-b);
end
semilogy(err);
```

well designed Q.

(b) Assume that we perform in total m=30 iterations of flexible GMRES. Now change the preconditioner in the previous script in a way that M(j,x) does m-j+1 steps of bicgstab. Does the situation change? Notice that the total complexity of the two script is the same.

(c) Now change the preconditioner in a way that M(j,x) does m steps of bicgstab (notice that this is still not a constant preconditioner). Does the situation change? Notice that the total complexity is now increased. Do we get any benefit in comparison with (b)?

(d) As you have noticed in the previous points (a) and (b), if we have the preconditioners $M_1, M_2, M_3, \ldots$ the order in which we use them is important. After the point (a) and (b), can you reach any conclusion? It is better to use "better" preconditioners at the first iterations and "worse" preconditioners at the last iterations or vice-versa? Do you think this is a general property? Test it more changing the matrix A and the preconditioners.

Note: this approach is not completely meaningless. Indeed, bcgstab require, as memory $O(n)$ independently on the number of iterations, whereas gmres requires $O(mn)$ memory. However, for many problems bcgstab may not work but still be a valid preconditioner and GMRES may require too many iterations (and memory) without the usage of a preconditioner.

Solution 1-21

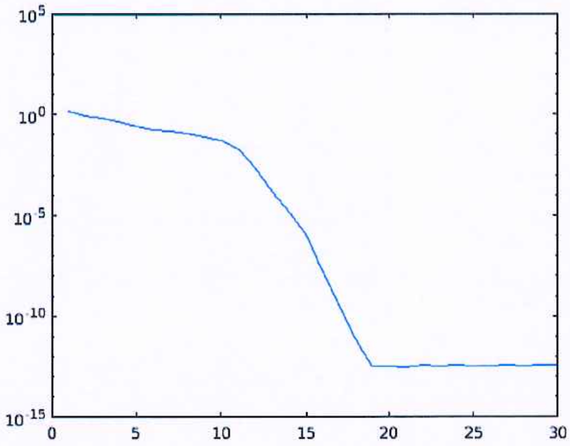
Solution by Emil:

A packed version of GMRES and FGMRES are available among the files for Block 1 and called my_GMRES and my_FGMRES, see the problem 1.23.

a) Modifying the given code in this problem we come up with the solution

```
clear; close all; clc;

n=200;
e = ones(n,1);
A = spdiags([e 2*e e], -1:1, n, n);

b=rand(n,1);
norm_b=norm(b);

V(:,1)=b/norm_b;

m=30;
TOL = -inf;
M=@(j,b) bicgstabl(A,b,TOL,j);

for j=1:m
 Z(:,j)=M(j,V(:,j));
 V(:,j+1)=A*Z(:,j);
 h=V(:,1:j)'*V(:,j+1);
 V(:,j+1)=V(:,j+1)-V(:,1:j)*h;
 g=V(:,1:j)'*V(:,j+1);
 V(:,j+1)=V(:,j+1)-V(:,1:j)*g;
 H(1:j,j)=h+g;
 H(j+1,j)=norm(V(:,j+1));
 V(:,j+1)=V(:,j+1)/H(j+1,j);

 e1=eye(j+1); e1=e1(:,1);
 y=H\(norm_b*e1);
 xx=Z(:,1:j)*y;
 err(j)=norm(A*xx-b);
end
semilogy(err);
```

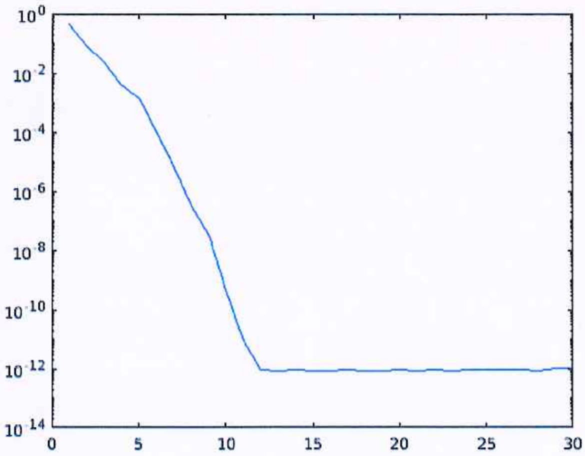
The convergence looks like:



b) The change in the script is minor. The only difference is that the preconditioner is now defined by

```
M=@(j,b) bicgstabl(A,b,TOL,m-j+1);
```

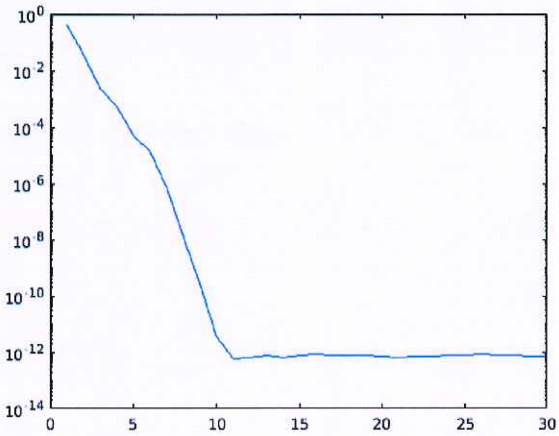
and the convergence looks like:



c) Once again the change to the code is minor, and the preconditioner is defined by

```
M=@(j,b) bicgstabl(A,b,TOL,m);
```

The convergence now becomes:



where we can notice that the change from b) is only minor, convergence is like one or two iterations faster.

d) A comparison with the two plots in a) and b) indicate that the convergence is improved by starting out with a better preconditioner, rather than using a good preconditioner in the end. I am not sure if this is (or at least if I can prove that it is) better to use a good preconditioner in the beginning, like in b), but I guess so. My intuition is that a good preconditioner will help capture the "good" subspace to search for the solution, and if that is done in a better way early in the process, then a "better" subspace will be built.

Further tests with with the FEM-matrix

```
nn = 50;
A = gallery('wathen',nn,nn);
n = size(A,1);
```

and the random matrix

```
n = 200;
rng(123456789)
A = gallery('uniformdata',n,n) - 4.5*eye(n,n);
```

yeild similar results.

Moderator comment: Correct!

Problem 1-22

Let A be a matrix such that

$$\sum_{j=0}^{m} A^j = I$$

with m small, and let $M = AP$. We want to solve $Mx = b$.

- If use GMRES with P as left preconditioner, in how many iterations will it converge?
- If use GMRES with P as right preconditioner, in how many iterations will it converge?
- If P is such that $P = Q^2$ and we use Q as split preconditioner, in how many iterations gmres will converge?
- Usually one reads in books (including Saad) that left and right preconditoner are practically equivalent, in the sense that if we use P as left or right preconditioner we obtain the same performance. Can you explain in which sense are they equivalent?
- (Optional) Write a matlab code for the first two points, including generating such matrix A with, e.g., $m = 4$.
- (Optional) Can you find an example where a matrix P works better as left preconditoner and worse as right preconditioner (or in general it is more beneficial to use it in that way)? (This does not contradict the previous point). No simulation required, it is enough to present an argument.

Solution 1-22

Problem 1-23

Problem by Emil:

This is a (slightly artificial) problem about Flexible GMRES (FGMRES) and the change of preconditioners. It is inspired by problem 4.12 in the Numerical Linear Algebra course.

a) Consider the MATLAB code:

```
n = 100;
TOL = 1e-11;
MAXIT = 30;

h = 1/(n+1);
Dxx = spdiags((1/h^2) * [1*ones(n,1), -2*ones(n,1), 1*ones(n,1)], [-1,0,1], n, n);
I = speye(n);

f = @(x,y) abs(x-y);
g = @(a,x,y) a*((x-1/2).^2+(y-1/2).^2).^(1/2);

x = linspace(h,1-h, n);
y = linspace(h,1-h, n);
[X, Y] = meshgrid(x, y);

G = g(1.56, X, Y);
A = kron(Dxx, I) + kron(I, Dxx) + spdiags(G(:), 0, n^2, n^2);
F = f(X, Y);
b = F(:);

prec = @(C) lyap(Dxx, -C);
apply_prec_1 = @(x) reshape(prec(reshape(x, n, n)), n^2, 1);
apply_prec_2 = @(x) reshape(prec(reshape(x, n, n)) + 1e-10*symmpart(rand(n,n)), n^2, 1);

[x_gmres1, r_norm_gmres1, error_gmres1] = my_GMRES(A, b, TOL, MAXIT, apply_prec_1);
[x_gmres2, r_norm_gmres2, error_gmres2] = my_GMRES(A, b, TOL, MAXIT, apply_prec_2);
```

This is applying GMRES to solve a discretized PDE: $z_{xx}(x,y) + z_{yy}(x,y) + g(x,y) \cdot z(x,y) = f(x,y)$, and preconditioned with solving the Lyapunov equation, i.e., the discretized version of $z_{xx}(x,y) + z_{yy}(x,y) = k(x,y)$. However, for the second preconditioner, this one is artificially disturbed with a random noise. What can be observed here? How big/small does the disturbance be? Plot the convergence in relative residual and in relative error.

b) Describe the changes that needs to be done to convert GMRES to FGMRES. Use this to make similar convergence plots. What can we see? How big/small can the disturbance be? What can be the reason?

Hint: Consider the eigenvalues of the preconditioned matrix $\sigma(M^{-1}A) = \sigma((M^{-1}A)^T) = \sigma(AM^{-1})$

c) Test with other functions f and g.

d) Test with a perturbed preconditioner of the type

```
prec2 = @(C) lyap(Dxx + 1e-2*symmpart(rand(n,n)), -C);
apply_prec_2 = @(x) reshape(prec2(reshape(x, n, n)), n^2, 1);
```

Note: The functions my_GMRES and my_FGMRES are provided in the files:
https://kth.instructure.com/groups/33520/files/folder/block1

Solution 1-23

Problem 1-7

Flexible GMRES question:

Equation (1) in Saad's paper states

$$AZ_m = V_m\overline{H}_m$$

Prove that this relation reduces to the standard Arnoldi relation if $M_j = M$, i.e., M_j is constant.

Solution 1-7

By construction of the algorithm we have that $z_j = M_j v_j$ (see Saad or Problem 1-21). Therefore, if $M_j = M$ we can write $Z_m = MV_m$. By replacing this expression in the Arnoldi-like factorization, we get

$$AZ_m = V_{m+1}\underline{H}_m$$

$$AMV_m = V_{m+1}\underline{H}_m$$

Observe that this is an Arnoldi factorization for the matrix AM.

Moderator comment: Correct!

Problem 1-38

Proposition 2.1 in the Saad's flexible GMRES-paper does not have a proof (and is expected to follow "trivially" from the above reasoning which is debatable). Provide the details, i.e., prove the following.

Proposition 2.1: The approximate solution x_m obtained at step m minimizes the residual norm

$$||b - Ax||_2$$

over

$$x \in x_0 + \text{Span}(Z)$$

(You may assume x_0=0 to see analogy with our standard GMRES-bounds.)

Solution 1-38

Problem 1-2

Setup a sparse linear system of equations to use as benchmark problem using COMSOL.

Examples and comsol-multiphysics files:

https://www.comsol.com/model/12385

https://www.comsol.com/model/8577

https://www.comsol.com/model/3576

https://www.comsol.com/model/847

Moderator comment: This problem can help us understand the performance of the preconditioners we use later. So please provide downloadable easy-to-use mat-files which do not require comsol. Hopefully we will be able to do better than the solvers in COMSOL.

Solution 1-2

Here is the procedure to export to MAT-files:

At a KTH-desktop computer you need to start the LiveLink framework of COMSOL (for communication with MATLAB-process) by running:

```
$ module add comsol
$ comsol server
Username: Myname
Password: Not important
Repeat password: Not important
```

Username and password can be chosen as you like.

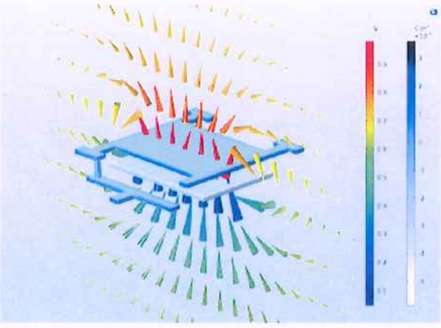
Start MATLAB and run

```
>> addpath('/opt/comsol/5.3/mli');
>> mphstart
>> mphlaunch;
>> model=mphopen('blabla.mph')
>> MM=mphmatrix(model,'sol1','out',{'K'});
>> size(MM.K)
>> K=MM.K;
>> save('/tmp/myfile.mat','K')
```

The matrix (K-matrix) of the MEMS device described here (https://www.comsol.com/model/tunable-mems-capacitor-123) can generated following the procedure above. Here is the discretized matrix:

capacitor_tunable_piezo_domain_K.mat

Here is the solution visualized in COMSOL:



A slight variation applied to the heat conductor problem gave:

disk_stack_heat_sink_52a.mat

Problem 1-12

A nonlinear preconditioner

We want to solve the linear systems $Ax = b$ with the preconditioner $M(x)$. We have always assumed that the preconditioner is given as a linear function $M(x)$ such that $M(x) \approx A^{-1}x$. More precisely we assumed that there is a matrix $P \approx A^{-1}$ such that $M(x) = Px$. In this exercise we will try to softer this assumption.

(a) Assume that $M(x) = \dfrac{Px}{\|Px\|}$, where P is a matrix. Observe that $M(x)$ is a nonlinear function. Can we use GMRES with this preconditioner? Can we use flexible GMRES instead? (This would make sense if in our application we know that the solution has norm one).

(b) Assume that $M(x) = \alpha(x)Px$ where $\alpha : \mathbb{R}^n \to \mathbb{R}$ is a nonlinear function and P is a matrix. Can we use GMRES with this preconditioner? Can we use flexible GMRES instead?

(c) Following the previous point, prove that, similar to Problem 1-7, the Arnoldi-like factorization $AZ_m = V_{m+1}\underline{H}_m$reduces to $APD_mV_m = V_{m+1}\underline{H}_m$ where D_m is a diagonal matrix.

(d) [Optional] Which other nonlinear functions $M(x)$ we can use as preconditioner? Give other examples.

Solution 1-12

Problem 1-16

This problem concerns Example 12.1 (page 107) in the publicly available teaching literature http://www.asc.tuwien.ac.at/~winfried/teaching/106.079/SS2017/downloads/iter.pdf

a) The preconditioner has to be faster computed than the actual (exact) inverse. Why is this the case in this preconditioner? (This is a small problem so it does not matter so much but for larger problems the same technique can be improved.)

b) Run the problem for a random RHS. Is it faster? Why?

c) This is a small problem, plot the pseudospectra of the original problem and the preconditioned problem and explain why the preconditioner works.

d) (Optional/fun) The author of the document has put the legend on top of some of the data in the figure, which violates the first rule of plotting, what matlab-commands moves it? (Or do it in julia)

Another problem with the plot in the example is the use of a log-log scale. It is easier to identify properties of the method in a semilogy plot, instead of a log-log.

Solution 1-16

Problem 1-31

Consider solving the linear system $\boldsymbol{Ax} = \boldsymbol{b}$ with the preconditioned conjugate gradient method of Algorithm 9.1 (Saad, Chapter 9) with the symmetric positive definite preconditioner $\boldsymbol{M} = \boldsymbol{L}\boldsymbol{L}^T$, where $\boldsymbol{L}$ is the Choleksy factor.

Show that this is equivalent to solving the preconditioned system

$$\boldsymbol{L}^{-1}\boldsymbol{A}\boldsymbol{L}^{-T}\boldsymbol{u} = \boldsymbol{L}^{-1}\boldsymbol{b}$$

where $\boldsymbol{u} = \boldsymbol{L}^T\boldsymbol{x}$ with the conjugate gradient method.

Solution 1-31

Solution by Federico:

The preconditioned CG has the following structure:

```
$r_0 = b - Ax_0$
$z_0 = M^{-1}r_0$
$p_0 = z_0$
for $j = 1, 2, \dots$
    $\alpha_j = \frac{(r_j, z_j)}{(Ap_j, p_j)}$
    $x_{j+1} = x_j + \alpha_j p_j$
    $r_{j+1} = r_j - \alpha_j A p_j$
    $z_{j+1} = M^{-1} r_{j+1}$
    $\beta_j = \frac{(r_{j+1}, z_{j+1})}{(r_j, z_j)}$
    $p_{j+1} = z_{j+1} + \beta_j p_j$
end
```

but we now consider $M = LL^T$ where L is the Choleski factor. We explicitate M in the expression for z_j:

$$z_{j+1} = M^{-1}r_{j+1} = (LL^T)^{-1}r_{j+1} = L^{-T}L^{-1}r_{j+1} = L^{-T}\tilde{r}_{j+1}$$

and from this we make explicit the inner products of β_j:

$$\beta_j = \frac{(r_{j+1}, z_{j+1})}{(r_j, z_j)} = \frac{(r_{j+1}, L^{-T}L^{-1}r_{j+1})}{(r_j, L^{-T}L^{-1}r_j)} = \frac{(L^{-1}r_{j+1}, L^{-1}r_{j+1})}{(L^{-1}r_j, L^{-1}r_j)} = \frac{(\tilde{r}_{j+1}, \tilde{r}_{j+1})}{(\tilde{r}_j, \tilde{r}_j)}$$

and of α_j: we want $(Ap_j, p_j) = (\tilde{A}\tilde{p}_j, \tilde{p}_j)$, where $\tilde{A} = L^{-1}AL^{-T}$.

$$(Ap_j, p_j) = (AL^{-T}L^Tp_j, L^{-T}L^Tp_j) = (L^{-1}AL^{-T}L^Tp_j, L^Tp_j) = (\tilde{A}\tilde{p}_j, \tilde{p}_j)$$

Now we can rewrite the steps of CG:

$\alpha_j = \frac{(\tilde{r}_j, \tilde{r}_j)}{(\tilde{A}\tilde{p}_j, \tilde{p}_j)}$, and then, by multiplying from the left by L^T,

$$L^Tx_{j+1} = L^Tx_j + \alpha_j L^Tp_j \;\Rightarrow\; \tilde{x}_{j+1} = \tilde{x}_j + \alpha_j\tilde{p}_j.$$

By multiplying from the left by L^{-1},

$$r_{j+1} = r_j - \alpha_j Ap_j \;\Rightarrow\; \tilde{r}_{j+1} = \tilde{r}_j - \alpha_j L^{-1}AL^{-T}L^Tp_j = \tilde{r}_j - \alpha_j\tilde{A}\tilde{p}_j$$

By multiplying from the left by L^T,

$$p_{j+1} = z_{j+1} + \beta_j p_j \;\Rightarrow\; \tilde{p}_{j+1} = L^TL^{-T}\tilde{r}_{j+1} + \beta_j\tilde{p}_j = \tilde{r}_{j+1} + \beta_j\tilde{p}_j$$

This is the CG corresponding to the system

$\tilde{A}\tilde{x} = L^{-1}b$, where $\tilde{x} = L^T x$. In fact the starting conditions is

$\tilde{r}_0 = L^{-1}r_0 = L^{-1}b - L^{-1}Ax_0 = L^{-1}b - L^{-1}AL^{-T}L^T x_0 = L^{-1}b - \tilde{A}\tilde{x}$

Problem 1-32

Show that the spectra of the operators corresponding to left, right and split preconditioning are identical. I.e. show that if the preconditioning matrix M is factorized as $M = LU$ then $M^{-1}A$, AM^{-1} and $L^{-1}AU^{-1}$ have the same eigenvalues. M non-singular

---

Solution 1-32

Solution by Parikshit:

Let us assume (λ, x) is an eigenpair of $M^{-1}A$. Then,

$M^{-1}Ax = \lambda x \Leftrightarrow AM^{-1}(Mx) = \lambda(Mx) \Leftrightarrow Ax = \lambda LUx \Leftrightarrow L^{-1}AU^{-1}(Ux) = \lambda(Ux)$

Hence, $M^{-1}A$, AM^{-1} and $L^{-1}AU^{-1}$ have identical eigenvalues with eigenvectors x, Mx and Ux respectively.

Problem 1-33

Compare right versus left preconditioned GMRES.

a) Describe the most relevant differences, e.g. what residual norm is to be minimized, difference in generated Krylov subspace etc..

b) The spectra of the operators $M^{-1}A$ and $AM^{-1}$are identical and hence their convergence behavior should be similar according to convergence theory based on eigenvalues. However there is still situations when there could be a substantial differences in convergence behavior. Give at least one example of such a situation.

Solution 1-33

Problem 1-37

Problem by Parikshit:

Assume that we are applying a Krylov subspace procedure for the system $Ax = b$ using a preconditioner M which is related to A by the following relation: $M = A - R$, where R has significantly less non-zero entries compared to A. At each step of the iteration, we must perform the transformation $w = M^{-1}Av$ Given that A and M are related in a specific way, is there a more efficient way to perform this transformation without computing Av explicitly at each step?

Solution 1-37

Problem 1-39

Solution 1-39