FEM and sparse linear system solving

Lecture 9, Nov 17, 2017: Krylov space methods

http://people.inf.ethz.ch/arbenz/FEM17

Peter Arbenz
Computer Science Department, ETH Zürich
E-mail: arbenz@inf.ethz.ch

Survey on lecture

- The finite element method
- Direct solvers for sparse systems
- Iterative solvers for sparse systems
  - Stationary iterative methods, preconditioning
  - Steepest descent and conjugate gradient methods
  - Krylov space methods for nonsymmetric systems
    GMRES, MINRES
  - Preconditioning
  - Nonsymmetric Lanczos iteration based methods
    Bi-CG, QMR, CGS, BiCGstab
  - Multigrid (preconditioning)

Outline of this lecture

1. Krylov (sub)spaces
2. Orthogonal bases for Krylov spaces
3. GMRES
4. MINRES

Krylov space methods are 'matrix-free':
only a MatVec function is necessary, that computes $\boldsymbol{y} \leftarrow A\boldsymbol{x}$.

Krylov spaces

In steepest descent and CG we have

$$\boldsymbol{r}_k = \boldsymbol{r}_{k-1} - \alpha_{k-1} A\boldsymbol{p}_{k-1}, \qquad \boldsymbol{p}_k = \boldsymbol{r}_k + \beta_{k-1}\boldsymbol{p}_{k-1},$$

where $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k$ is the residual and $\boldsymbol{p}_k$ is the search direction. Since $\boldsymbol{p}_k$ depends on $\boldsymbol{p}_{k-1}$ and $\boldsymbol{r}_k$ we can write

$$\boldsymbol{r}_k = \boldsymbol{r}_0 + \sum_{j=1}^{k} c_j A^j \boldsymbol{r}_0 \quad \Longleftrightarrow \quad \boldsymbol{r}_k = p_k(A)\boldsymbol{r}_0, \quad p_k(0) = 1,$$

Thus, the residuals of these methods can be written as a k-th order polynomial in A applied to the initial residual $\boldsymbol{r}_0$.

Krylov spaces (cont.)

Moreover,

$$\sum_{j=1}^{k} c_j A^j \boldsymbol{r}_j = \boldsymbol{r}_k - \boldsymbol{r}_0$$

$$= (\boldsymbol{b} - A\boldsymbol{x}_k) - (\boldsymbol{b} - A\boldsymbol{x}_0) = -A(\boldsymbol{x}_k - \boldsymbol{x}_0)$$

such that

$$\boldsymbol{x}_k = \boldsymbol{x}_0 - \sum_{j=0}^{k-1} c_{j+1} A^j \boldsymbol{r}_0.$$

Krylov spaces (cont.)

Definition *For given A, the m-th Krylov space generated by the vector $\boldsymbol{r}$ is given by*

$$\mathcal{K}_m = \mathcal{K}_m(A, \boldsymbol{r}) := span\left\{\boldsymbol{r}, A\boldsymbol{r}, A^2\boldsymbol{r}, \ldots, A^{m-1}\boldsymbol{r}\right\}.$$

We can also write

$$\mathcal{K}_m(A, \boldsymbol{r}) = \left\{p(A)\boldsymbol{r} \mid p \in \mathbb{P}_{m-1}\right\}.$$

where $\mathbb{P}_d$ denotes the set of polynomials of degree at most d.

For the residuals $\boldsymbol{r}_m$ of stationary or CG iterations we have

$$\boldsymbol{r}_m \in \boldsymbol{r}_0 + A\mathcal{K}_{m-1}(A, \boldsymbol{r}_0) \subset \mathcal{K}_m(A, \boldsymbol{r}_0).$$

Cayley–Hamilton theorem

Theorem: Let $\chi_A(\zeta) := \det(\zeta I - A)$ denote the *characteristic polynomial* of $A \in \mathbb{R}^{n\times n}$. Then

$$\chi_A(A) = O.$$

Corollary: Let A be nonsingular and

$$\chi_A(\zeta) = \zeta^n + a_{n-1}\zeta^{n-1} + a_{n-2}\zeta^{n-2} + \cdots + a_0,$$

with $a_0 = \det(A) \neq 0$, then

$$A^{-1} = -\frac{1}{a_0}\left[A^{n-1} + a_{n-1}A^{n-2} + a_{n-2}A^{n-3} + \cdots + a_1 I\right].$$

Cayley–Hamilton theorem (cont.)

Consequence:

- If A nonsingular, then A^{-1} is a polynomial in A of degree at most $n-1$.
- If $A\boldsymbol{x} = \boldsymbol{b}$ then

$$\boldsymbol{x} = (A^{n-1}\boldsymbol{b} + a_{n-1}A^{n-2}\boldsymbol{b} + a_{n-2}A^{n-3}\boldsymbol{b} + \cdots + a_1\boldsymbol{b})/(-a_0).$$

- As $A\boldsymbol{e}_0 = \boldsymbol{r}_0$ we have

$$\boldsymbol{e}_0 = (A^{n-1}\boldsymbol{r}_0 + a_{n-1}A^{n-2}\boldsymbol{r}_0 + a_{n-2}A^{n-3}\boldsymbol{r}_0 + \cdots + a_1\boldsymbol{r}_0)/(-a_0).$$

- $\boldsymbol{x} \in \boldsymbol{x}_0 + \mathcal{K}_n(A, \boldsymbol{r}_0)$.

  Therefore it makes sense to look for approximate solutions in Krylov spaces.

Minimal polynomial

Definition

For $A \in \mathbb{R}^{n\times n}$, the unique monic polynomial μ_A of minimal degree d satisfying $\mu_A(A) = 0$ is called the *minimal polynomial* of A.

Theorem: Let A have distinct eigenvalues $\lambda_1, \ldots, \lambda_k$, then

$$\mu_A(\zeta) = \prod_{i=1}^{k} (\zeta - \lambda_i)^{r_i},$$

where r_i is the order of the largest Jordan block of A for λ_i.

Note:

- The degree of μ_A is at most n (Cayley–Hamilton).
- $\boldsymbol{e}_0 \in \mathcal{K}_d$, interesting when $d \ll n$
- Ideal: *few distinct* eigenvalues, *small* Jordan blocks!

Dependence on right-hand side

What does it mean when $\mathcal{K}_1 \subsetneqq \mathcal{K}_2 \subsetneqq \cdots \subsetneqq \mathcal{K}_p = \mathcal{K}_{p+1} = \cdots$?

$$A^p \boldsymbol{r}_0 = \alpha_0 \boldsymbol{r}_0 + \alpha_1 A \boldsymbol{r}_0 + \alpha_2 A^2 \boldsymbol{r}_0 + \cdots + \alpha_{p-1} A^{p-1} \boldsymbol{r}_0$$

Multiplying by A^{-1} and noting that $A^{-1}\boldsymbol{r}_0 = \boldsymbol{e}_0$ gives

$$A^{p-1} \boldsymbol{r}_0 = \alpha_0 \boldsymbol{e}_0 + \alpha_1 \boldsymbol{r}_0 + \alpha_2 A \boldsymbol{r}_0 + \cdots + \alpha_{p-1} A^{p-2} \boldsymbol{r}_0$$

Therefore $\boldsymbol{e}_0 \in \mathcal{K}_p(A, \boldsymbol{r}_0)$ or

$$\boldsymbol{x}^* \in \boldsymbol{x}_0 + \mathcal{K}_p(A, \boldsymbol{r}_0).$$

We get the solution after precisely p steps.

Note that p depends on $\boldsymbol{r}_0$ (and thus on $\boldsymbol{x}_0$).

Dependence on right-hand side (cont.)

Let A be diagonalizable with $A = U\Lambda U^{-1}$.
This means that there n linearly independent vectors (eigenvectors) with $A\boldsymbol{u}_i = \boldsymbol{u}_i\lambda_i$, $1 \leq i \leq n$.

If $\boldsymbol{r}_0 = \sum_{i=1}^{p} \alpha_i \boldsymbol{u}_i$ then $A^j\boldsymbol{r}_0 = \sum_{i=1}^{p} \alpha_i \lambda^j \boldsymbol{u}_i$.

So, if $\boldsymbol{r}_0 \in \text{span}\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_p\}$ with $A\boldsymbol{u}_i = \boldsymbol{u}_i\lambda_i$, then $\mathcal{K}_p = \mathcal{K}_{p+1}$ means that the solution is found in p iteration steps.

This is an uncommon situation! Usually, $p \approx n$ and it is not practical to iterate until the Krylov space is exhausted.
We try to get a good approximation of $\boldsymbol{x}^*$ after $m \ll n$ iteration steps.

What is the best x_m from $\mathcal{K}_m$?

Want to *best* approximate $\boldsymbol{x}^* = A^{-1}\boldsymbol{b}$ from $\boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$.
Various answers lead to different methods:

- Ideal but not practical: minimize $\|\boldsymbol{e}_m\|_2 = \|\boldsymbol{x}^* - \boldsymbol{x}_m\|_2$.
- When A is SPD, then we can minimize $\|\boldsymbol{e}_m\|_A = \|\boldsymbol{r}_m\|_{A^{-1}}$. *Conjugate Gradients*
- Can minimize $\|\boldsymbol{r}_m\|_2$. For symmetric A: *MINimum RESidual*, for nonsymmetric A: *Generalized Minimum RESidual*
- Can enforce Galerkin condition $\boldsymbol{r}_m \perp \mathcal{K}_m$ or Petrov-Galerkin condition $\boldsymbol{r}_m \perp \mathcal{L}_m$, e.g. *BiConjugate Gradients* and *Quasi Minimum Residual*, these are m constraints to compute m parameters $\alpha_0, \ldots, \alpha_{m-1}$ from
  $\boldsymbol{x}_m = \boldsymbol{x}_0 + \alpha_0\boldsymbol{r}_0 + \alpha_1 A\boldsymbol{r}_0 + \alpha_2 A^2\boldsymbol{r}_0 + \cdots + \alpha_{m-1}A^{m-1}\boldsymbol{r}_0$.

An orthogonal basis for $\mathcal{K}_m$

Problem: The matrix

$$K_m(A, \boldsymbol{r}_0) := \left[\begin{array}{c|c|c|c} & & & \\ \boldsymbol{r}_0 & A\boldsymbol{r}_0 & \cdots & A^{m-1}\boldsymbol{r}_0 \\ & & & \end{array} \right]$$

becomes more and more ill-conditioned as k increases. (Remember vector iteration for computing largest eigenvalue.)

Solution: We have to find a well-conditioned basis of $\mathcal{K}_m$.

Arnoldi & Lanczos algorithms

Task: For $j = 1, 2, \ldots, m$, compute orthonormal bases $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_j\}$ for the Krylov spaces

$$\mathcal{K}_j = \text{span}\left\{\boldsymbol{r}_0, A\boldsymbol{r}_0, A^2\boldsymbol{r}_0, \ldots, A^{j-1}\boldsymbol{r}_0\right\}.$$

The algorithms that do this are

- Lanczos algorithm for A symmetric/Hermitian.
- Arnoldi algorithm for A nonsymmetric.

In principle, Lanczos and Arnoldi algorithms implement the Gram–Schmidt orthogonalization procedure.

Difficulty: Because of ill-conditioning, we do not want to *explicitly* form $\boldsymbol{r}_0, A\boldsymbol{r}_0, \ldots, A^j\boldsymbol{r}_0$.

Arnoldi & Lanczos algorithms (cont.)

Instead of using $A^j \boldsymbol{r}_0$ we proceed with $A\boldsymbol{v}_j$.
(Notice that $A\boldsymbol{v}_i \in \mathcal{K}_{i+1} \subset \mathcal{K}_j$ for all $i < j$.)

Orthogonalize $A\boldsymbol{v}_j$ against $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_j$ by the Gram–Schmidt procedure:

$$\boldsymbol{w}_j = A\boldsymbol{v}_j - \sum_{i=1}^{j} \boldsymbol{v}_i h_{ij}.$$

$\boldsymbol{w}_j$ points in the desired new direction (unless it is $\mathbf{0}$). Therefore,

$$\boldsymbol{v}_{j+1} = \boldsymbol{w}_j / \|\boldsymbol{w}_j\|.$$

Q: What happens if $\boldsymbol{w}_j = \mathbf{0}$?

Arnoldi: Classical and modified Gram–Schmidt

$\boldsymbol{v}_1 = \boldsymbol{r}_0/\|\boldsymbol{r}_0\|_2$
for j:=1 to m **do**
  $\boldsymbol{w}_j = A\boldsymbol{v}_j$;
  for i:=1 to j **do**
    Orthogonalize $A\boldsymbol{v}_j$ against $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_j$:
    $\begin{cases} h_{i,j} := (A\boldsymbol{v}_j)^* \boldsymbol{v}_i; & \text{(Classical Gram–Schmidt)} \\ h_{i,j} := \boldsymbol{w}_j^* \boldsymbol{v}_i; & \text{(Modified Gram–Schmidt)} \end{cases}$
    $\boldsymbol{w}_j := \boldsymbol{w}_j - h_{i,j} * \boldsymbol{v}_i$;
  end for
  $h_{j+1,j} = \|\boldsymbol{w}_j\|_2$. (Stop if $h_{j+1,j}$ is zero.)
  $\boldsymbol{v}_{j+1} = \boldsymbol{w}_j/h_{j+1,j}$;
end for

In practice: MGS is *more* numerically stable than CGS

Remark: Classical vs. modified Gram–Schmidt

Given vectors $\boldsymbol{a}$ and $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_j$ with $\boldsymbol{v}_k^T \boldsymbol{v}_\ell = 0$ for all $1 \leq k, \ell \leq j$.

Task: Compute $\boldsymbol{z}$ s.t. $\boldsymbol{a} = \sum_i \alpha_i \boldsymbol{v}_i + \boldsymbol{z}$ with $\boldsymbol{z}^T \boldsymbol{v}_\ell = 0$ for all ℓ.

Classical Gram–Schmidt

$\boldsymbol{z} = \boldsymbol{a}$;
for i:=1 to j **do**
  $\alpha_i = \boldsymbol{v}_i^T \boldsymbol{a}$;
  $\boldsymbol{z} = \boldsymbol{z} - \boldsymbol{v}_i \alpha_i$;
end for

Modified Gram–Schmidt

$\boldsymbol{z} = \boldsymbol{a}$;
for i:=1 to j **do**
  $\alpha_i = \boldsymbol{v}_i^T \boldsymbol{z}$;
  $\boldsymbol{z} = \boldsymbol{z} - \boldsymbol{v}_i \alpha_i$;
end for

MGS is more stable than CGS.

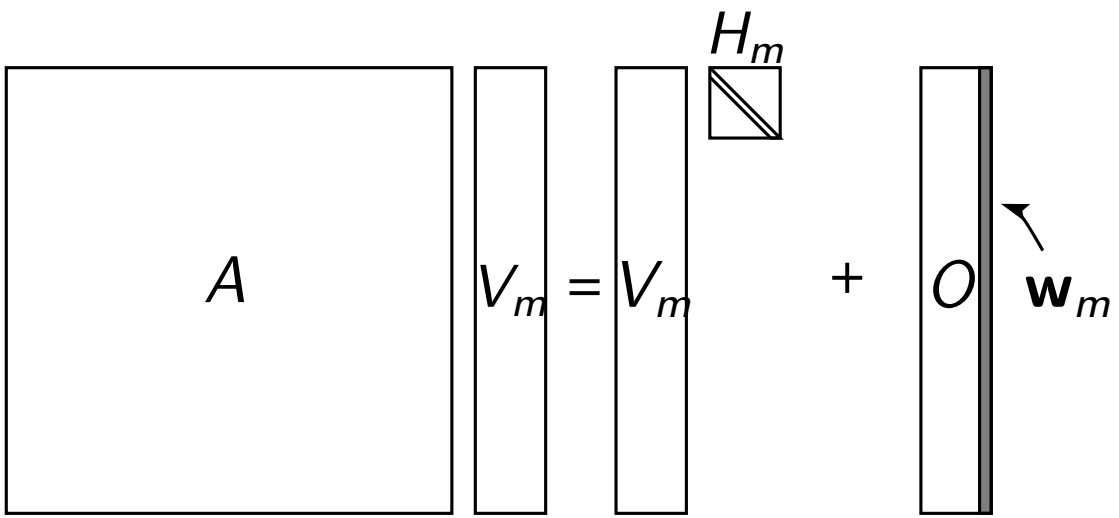
There are other solutions as, e.g., Householder reflectors.

See Golub–van Loan: Matrix Computations.

Arnoldi relation

Define $V_m := [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m]$. Then we get the Arnoldi relation

$$AV_m = V_m H_m + \boldsymbol{w}_m \boldsymbol{e}_m^T = V_{m+1} \bar{H}_m.$$



Arnoldi relation (cont.)

Here,

$$\bar{H}_m = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1,m} \\ h_{21} & h_{22} & \cdots & h_{2,m} \\ & h_{3,2} & \cdots & h_{3,m} \\ & & \ddots & \vdots \\ & & & h_{m+1,m} \end{bmatrix} \in \mathbb{R}^{(m+1)\times m}$$

The square matrix $H_m \in \mathbb{R}^{m\times m}$ is obtained from $\bar{H}_m$ by deleting the last row.

Notice that

$$H_m = V_m^T A V_m.$$

Therefore, if A is symmetric $\Rightarrow H_m \equiv T_m$ is tridiagonal!

The Lanczos relation is $AV_m = V_m T_m + \boldsymbol{w}_m \boldsymbol{e}_m^T = V_{m+1} \overline{T}_m$.

GMRES

The GMRES (Generalized Minimal Residual) algorithm computes $\boldsymbol{x}_m \in \boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$ that leads to the smallest residual exploiting the Arnoldi relation.
The MINRES algorithm does the same for symmetric matrices using the Lanczos relation.

Goal: Cheaply minimize

$$\|\boldsymbol{r}_m\|_2 = \|\boldsymbol{b} - A\boldsymbol{x}_m\|_2, \qquad \boldsymbol{x}_m \in \boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0),$$

using the Arnoldi relation

$$AV_m = V_{m+1}\bar{H}_m$$

and $\mathcal{R}(V_m) = \mathcal{K}_m(A, \boldsymbol{r}_0)$.

A Hessenberg Least-Squares problem

Arnoldi basis expansion $\boldsymbol{x}_m = \boldsymbol{x}_0 + V_m \boldsymbol{y}$ gives with $\beta := \|\boldsymbol{r}_0\|_2$:

$$\begin{aligned}
\min \|\boldsymbol{r}_m\|_2 &= \min_{\boldsymbol{y}} \|\boldsymbol{b} - A(\boldsymbol{x}_0 + V_m \boldsymbol{y})\|_2 \\
&= \min_{\boldsymbol{y}} \|\boldsymbol{r}_0 - A V_m \boldsymbol{y}\|_2 \\
&= \min_{\boldsymbol{y}} \|\boldsymbol{r}_0 - V_{m+1} \bar{H}_m \boldsymbol{y}\|_2 \\
&= \min_{\boldsymbol{y}} \| V_{m+1} \left(\beta \boldsymbol{e}_1 - \bar{H}_m \boldsymbol{y}\right) \|_2 \\
&= \min_{\boldsymbol{y}} \|\beta \boldsymbol{e}_1 - \bar{H}_m \boldsymbol{y}\|_2 .
\end{aligned}$$

- Hessenberg least squares problem of dimension $(m+1) \times m$.
- QR factorization of $\bar{H}_m$ can be cheaply computed using Givens rotations.

QR factorization of $\bar{H}_m$ using m Givens row rotations

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ & h_{32} & h_{33} & h_{34} \\ & & h_{43} & h_{44} \\ & & & h_{54} \end{bmatrix} \Rightarrow \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & r'_{14} \\ 0 & r'_{22} & r'_{23} & r'_{24} \\ & h_{32} & h_{33} & h_{34} \\ & & h_{43} & h_{44} \\ & & & h_{54} \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & r'_{14} \\ 0 & r''_{22} & r''_{23} & r''_{24} \\ & 0 & r'_{33} & r'_{34} \\ & & h_{43} & h_{44} \\ & & & h_{54} \end{bmatrix} \Rightarrow \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & r'_{14} \\ 0 & r''_{22} & r''_{23} & r''_{24} \\ & 0 & r''_{33} & r''_{34} \\ & & 0 & r'_{44} \\ & & & h_{54} \end{bmatrix} \Rightarrow \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & r'_{14} \\ 0 & r''_{22} & r''_{23} & r''_{24} \\ & 0 & r''_{33} & r''_{34} \\ & & 0 & r''_{44} \\ & & & 0 \end{bmatrix}$$

Apply same rotations to rhs & solve triangular $m \times m$ system:

$$\min \|\boldsymbol{r}_m\|_2 = \min \|\beta \boldsymbol{e}_1 - \bar{H}_m \boldsymbol{y}\|_2 = \min \|\beta \boldsymbol{z} - \bar{R}_m \boldsymbol{y}\|_2.$$

Residual results

Monotonicity: $\|\boldsymbol{r}_{m+1}\|_2 \leq \|\boldsymbol{r}_m\|_2$ because $\mathcal{K}_m \subset \mathcal{K}_{m+1}$.

Residual polynomial: Since $\boldsymbol{r}_m = \boldsymbol{b} - A\boldsymbol{x}_m \in \boldsymbol{r}_0 + A\mathcal{K}_m$, we have $\boldsymbol{r}_m = p_m(A)\boldsymbol{r}_0$.
p_m is a polynomial of degree m and $p_m(0) = 1$. Denote the set of all such polynomials by $\mathbb{P}'_m$.

Theorem
Let $A = Q\Lambda Q^{-1}$ be diagonalizable. Then at step m of the GMRES iteration, the residual $\boldsymbol{r}_m$ satisfies

$$\frac{\|\boldsymbol{r}_m\|_2}{\|\boldsymbol{r}_0\|_2} \leq \inf_{p_m \in \mathbb{P}'_m} \|p_m(A)\|_2 \leq \kappa(Q) \inf_{p_m \in \mathbb{P}'_m} \max_{\lambda \in \sigma(A)} |p_m(\lambda)|.$$

Residual results (cont.)

- We have seen a similar result for the CG algorithm. There $\kappa(Q) = 1$ and $\sigma(A) \subset [\alpha, \beta]$, $\alpha > 0$.
- In more general situations, the distribution of A's eigenvalues is more complicated.
- One may replace $\sigma(A)$ by a simpler shaped set $D \supset \sigma(A)$ to simplify the analysis. E.g., D may be an ellipse for which more general Chebyshev polynomials have been developed.
- But 0 must never be contained in D!

Sample matrices & convergence

Remember: minimal polynomial μ_A, assume that
A diagonalizable $\Leftrightarrow$ all Jordan blocks size one

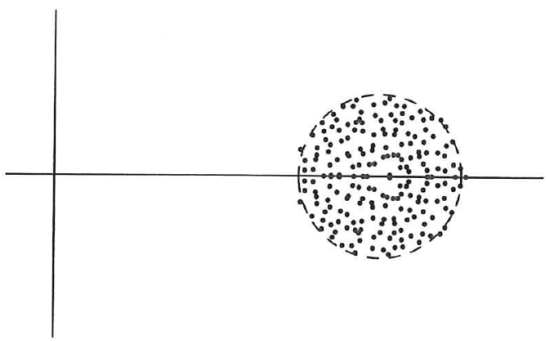
- Ideally: *few distinct* eigenvalues
- Heuristically: eigenvalues should best be 'squeezed' in small ball
- Perfect: A is (multiple of) identity: one point spectrum
- $\Rightarrow$ Visual interpretation of preconditioning goal

Compare convergence of GMRES for two matrices with quite different spectra

Pictures from Trefethen–Bau, *Numerical Linear Algebra*, SIAM 1997, Lecture 35.

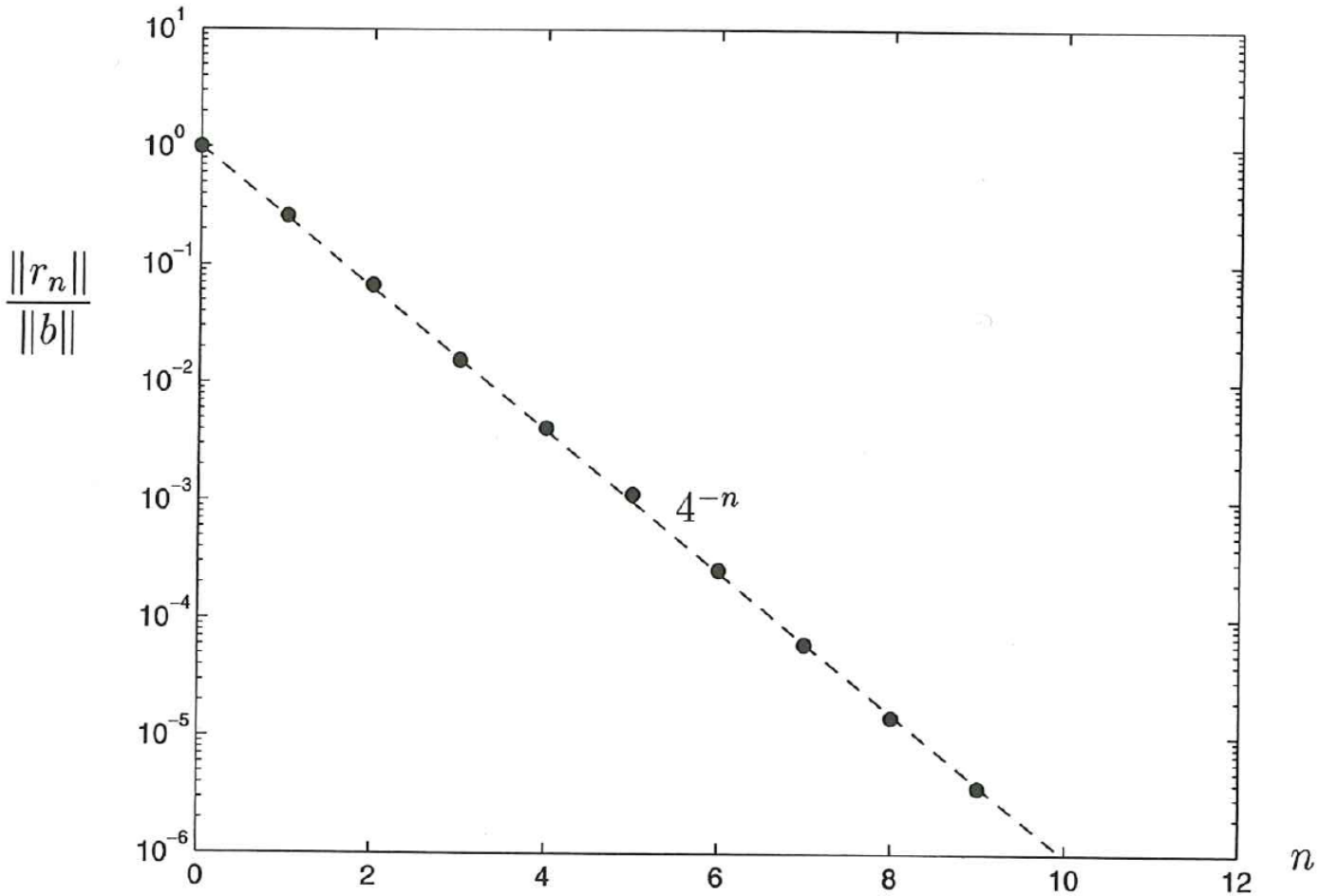
Matrix 1 – spectrum

Matrix A is obtained in MATLAB by
`m = 200; A = 2*eye(m) + 0.5*randn(m)/sqrt(m);`.



- A has a spectrum in a ball with radius $1/2$ centered at 2
- Let us choose $p_n(z) = (1 - z/2)^n$.
- On the spectrum $|1 - z/2| \leq 1/4$ and therefore $|p_n(z)| \leq (1/4)^n$.
- The convergence in this case is extraordinarily steady at a rate $\approx 4^{-n}$, see next slide.

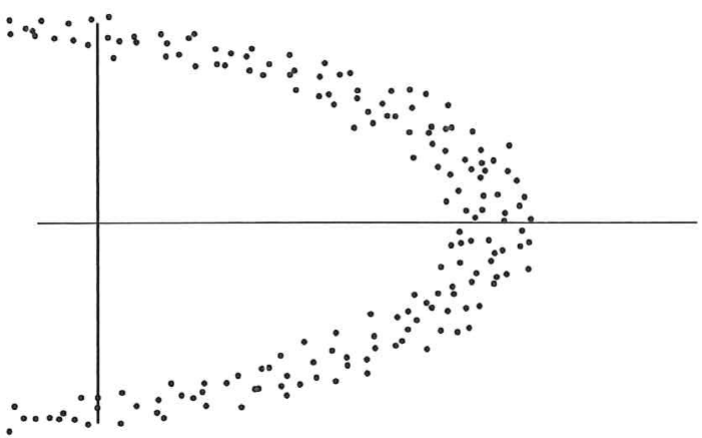
Matrix 1 – GMRES convergence



Matrix 2 - spectrum

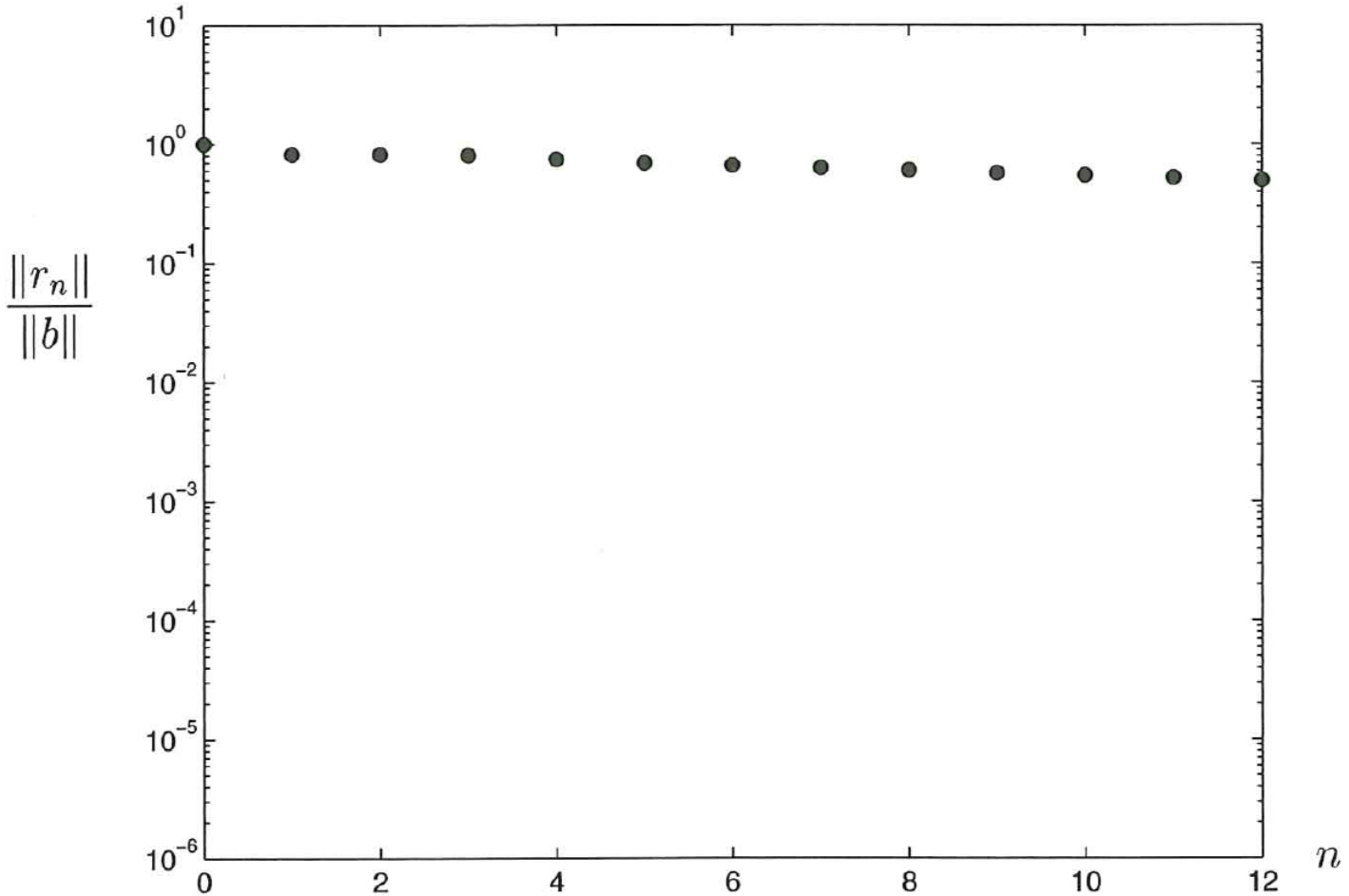
$A' = A + D$ where A is matrix 1 and D is the diagonal matrix with

$$d_k = (-2+2\sin\theta_k)+i\cos\theta_k, \qquad \theta_k = k\pi/(m+1), \qquad 0 \le k < m.$$



- A' has a spectrum that is 'surrounding the origin'. The eigenvalues lie in a semicircular cloud that bends around the origin.
- The convergence rate is much worse than in example 1.
- The condition of Q is not large.
- The slow convergence is due to the location of the eigenvalues.

Matrix 2 – GMRES convergence



Practical issues

- The GMRES algorithm with m steps needs memory space for $m + \mathcal{O}(1)$ vectors of length n plus the matrices A and M. CG needed just 4 vectors
- Slow convergence may entail expensive memory requirements. Recipe: limit m. Execute m steps of GMRES. Then restart with $\boldsymbol{x}_m$ as initial approximation of the next GMRES cycle.
- GMRES does not care about symmetries. There is no gain in choosing a symmetric preconditioner M when solving

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

- Incomplete LU factorizations (ILU) are popular preconditioners.

The GMRES(m) algorithm

Set an initial guess $\boldsymbol{x}_0$
until convergence **do**
    Compute initial residual, $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$
    Solve for preconditioned residual, $M\boldsymbol{z}_0 = \boldsymbol{r}_0$
    Normalize residual: $\boldsymbol{v}_1 = \boldsymbol{z}_0 / \|\boldsymbol{z}_0\|$
    for $k = 1, 2, ..., \boxed{m}$ **do**
        Compute matrix-vector product $\boldsymbol{r}_k = A\boldsymbol{v}_k$
        Solve $M\boldsymbol{z}_k = \boldsymbol{r}_k$
        Orthonormalize $\boldsymbol{z}_k$ with respect to all $\boldsymbol{v}_j$, $j = 1, \ldots, k$
        by modified Gram-Schmidt procedure to obtain $\boldsymbol{v}_{k+1}$
    end
    Solve GMRES minimization problem. Copy solution $\boldsymbol{x}_m$ into $\boldsymbol{x}_0$.
    Check convergence, if satisfied leave GMRES.
end

Krylov space methods for symmetric systems

Let A be symmetric indefinite. The CG method does not work since A does not induce a norm: $\boldsymbol{x}^T A\boldsymbol{x}$ and $\boldsymbol{x}^T A^{-1}\boldsymbol{x}$ can be negative and it does not make sense to minimize $\|\boldsymbol{r}\|_{A^{-1}}$.

Then the minimal residual (MINRES) algorithm is a feasible solution method.

MINRES is essentially GMRES adapted to symmetric systems. Two things are worth noting:

- $A\boldsymbol{v}_k$ does not need to be orthogonalized against $\boldsymbol{v}_j$ for $j < k-1$:

$$(A\boldsymbol{v}_k)^T\boldsymbol{v}_j = \boldsymbol{v}_k^T A^T\boldsymbol{v}_j \overset{A=A^T}{=} \boldsymbol{v}_k^T \underbrace{A\boldsymbol{v}_j}_{\in\mathcal{K}_{j+1}} = 0 \qquad \text{if } j+1 < k.$$

  We have a short recurrence.

Krylov space methods for symmetric systems (cont.)

- $\bar{H}_m =: \overline{T}_m$ is tridiagonal.

$$\bar{T}_m = \left[\begin{array}{ccccc} t_{11} & t_{12} & & & \\ t_{21} & t_{22} & t_{23} & & \\ & t_{3,2} & \ddots & \ddots & \\ & & \ddots & t_{m-1,m-1} & t_{m-1,m} \\ & & & t_{m,m-1} & t_{m,m} \\ \hline & & & & t_{m+1,m} \end{array}\right] = \left[\begin{array}{c} T_m \\ \hline t_{m+1,m}\boldsymbol{e}_m^T \end{array}\right]$$

Note that the submatrix T_m is symmetric.
Operating with Givens rotations on $\overline{T}_m$ zeros the lower off-diagonal elements at the expense of an additional upper off-diagonal. Of course the resulting matrix is upper-triangular.

The (unpreconditioned) MINRES algorithm

Choose $\boldsymbol{x}_0$.
Compute $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$. $\beta = \|\boldsymbol{r}_0\|$. Set $\boldsymbol{v}_0 = \boldsymbol{r}_0/\beta$.
for $k = 1, 2, \ldots$ **do**
  $\boldsymbol{w}_k := A\boldsymbol{v}_{k-1}$; **if** $k > 1$ **then** $\boldsymbol{w}_k = \boldsymbol{w}_k - \boldsymbol{v}_{k-2} t_{k,k-1}$
  $t_{k,k} := \boldsymbol{w}_k^T \boldsymbol{v}_{k-1}$; $\boldsymbol{w}_k = \boldsymbol{w}_k - \boldsymbol{v}_{k-1} t_{k,k}$
  $t_{k+1,k} := \|\boldsymbol{w}_k\|_2$; $\boldsymbol{y}_k = \boldsymbol{w}_k / t_{k+1,k}$;
  Update the QR factorization $\overline{T}_k = Q_k R_k$.
  Solve $R_k \boldsymbol{h}_k = \beta Q_k^T \boldsymbol{e}_1$.
  $\boldsymbol{x}_k := V_k \boldsymbol{h}_k$. $\boldsymbol{r}_k := \boldsymbol{b} - A\boldsymbol{x}_k$.
  Check for convergence.
end do

- The updates of $\boldsymbol{x}_k$ and $\boldsymbol{r}_k$ can be done by 3-term recurrencies.
- Preconditioners for A must be SPD in order to use this procedure. (Why?)

CG $\neq$ MINRES

In both methods: $\boldsymbol{x}_m \in \boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$

MINRES: Choose $\boldsymbol{x}_m$ such that $\|\boldsymbol{r}_m\|_2$ is minimized.

$$\min \|\boldsymbol{r}_m\|_2 = \min_{\boldsymbol{y}} \|\boldsymbol{r}_0 - AV_m\boldsymbol{y}\|_2 = \min_{\boldsymbol{y}} \|\beta\, \boldsymbol{e}_1 - \bar{T}_m\boldsymbol{y}\|_2.$$

CG: Choose $\boldsymbol{x}_m$ such that the Galerkin condition $\boldsymbol{r}_m \perp \mathcal{K}_m$ is satisfied.

$$\boldsymbol{0} = V_m^T(\boldsymbol{b} - A\boldsymbol{x}_m) = V_m^T(\boldsymbol{r}_0 - AV_m\boldsymbol{y}_m) = \beta\, \boldsymbol{e}_1 - T_m\, \boldsymbol{y}_m.$$

($\beta = \|\boldsymbol{r}_0\|_2$)

Main practical issues for Krylov space methods

- Select reasonable convergence criterion, and reasonable accuracy threshold τ.
- Without a good preconditioner, the method might *not* converge within a reasonable number of iteration steps.
- Cannot keep increasing the dimension of $\mathcal{K}$, not enough memory!
  $\Longrightarrow$ need to decide *how* and *when* to restart.