2024年秋季·多层迭代法

Krylov子空间方法

日期：2024.09.24

教师：张晨松，中国科学院数学与系统科学研究院

Section 1. Conjugate Gradient Method

Conjugate Directions

Now we consider a descent direction method with search direction $p^{(m)}$, i.e.

$$u^{(m+1)} = u^{(m)} + \alpha_m p^{(m)}.$$

In this case, the "optimal" stepsize from the exact line-search is

$$\alpha_m := \frac{\left(r^{(m)}, p^{(m)}\right)}{\left(p^{(m)}, p^{(m)}\right)_{\mathcal{A}}}.$$

We notice that the residual after one iteration is

$$r^{(m+1)} = r^{(m)} - \alpha_m \mathcal{A} p^{(m)}.$$

In order to keep the iteration going, we wish to construct a new search direction which is orthogonal to the previous search directions. This motives us to define

$$p^{(m+1)} := r^{(m+1)} + \beta_m p^{(m)}, \quad \text{such that} \quad \left(p^{(m)}, p^{(m+1)}\right)_{\mathcal{A}} = 0.$$

$$\Rightarrow \quad \beta_m := -\frac{\left(\mathcal{A} r^{(m+1)}, p^{(m)}\right)}{\left(\mathcal{A} p^{(m)}, p^{(m)}\right)}.$$

Properties of Conjugate Directions

Lemma (Properties of conjugate directions)

For any conjugate gradient step i, we have following identities:

1. $\left(r^{(i)}, p^{(i)}\right) = \left(r^{(i)}, r^{(i)}\right)$;
2. $\left(r^{(j)}, p^{(i)}\right) = 0, \qquad j > i$;
3. $\left(p^{(j)}, p^{(i)}\right)_{\mathcal{A}} = 0, \qquad j \neq i$;
4. $\left(r^{(j)}, r^{(i)}\right) = 0, \qquad j \neq i$.

Lemma (Stepsizes for CG)

For the conjugate gradient method, we have following identities:

$$\alpha_m = \frac{\left(r^{(m)}, r^{(m)}\right)}{\left(\mathcal{A}p^{(m)}, p^{(m)}\right)} \qquad \text{and} \qquad \beta_m = \frac{\left(r^{(m+1)}, r^{(m+1)}\right)}{\left(r^{(m)}, r^{(m)}\right)}.$$

Conjugate Gradient Method

Algorithm 2: Conjugate gradient method

```
1  %% Given an initial guess u and a tolerance ε;
2  r ← f − Au,  p ← r;
3  while ‖r‖ > ε
4        α ← (r, r)/(Ap, p);
5        ũ ← u + αp;
6        r̃ ← r − αAp;
7        β ← (r̃, r̃)/(r, r);
8        p̃ ← r̃ + βp;
9        Update: u ← ũ,  r ← r̃,  p ← p̃;
10 end
```

Lucky breakdown

Ref: Hestenes and Stiefel. "Methods of Conjugate Gradients for Solving Linear Systems". Journal of Research of the National Bureau of Standards 49, 1952

Convergence Rate of CG Method

Classical Results:

$$\left\|u-u^{(m)}\right\|_{\mathcal{A}} \leq 2\left(\frac{\sqrt{\kappa(\mathcal{A})}-1}{\sqrt{\kappa(\mathcal{A})}+1}\right)^{m}\left\|u-u^{(0)}\right\|_{\mathcal{A}}.$$

The residual $r^{(m)}$ is orthogonal to

$$\mathcal{K}_m = \text{span}\{r^{(0)}, \mathcal{A}r^{(0)}, \ldots, \mathcal{A}^{m-1}r^{(0)}\},$$

namely

$$(\mathcal{A}(u-u^{(m)}), v) = (r^{(m)}, v) = 0, \quad \forall v \in \mathcal{K}_m.$$

This implies

$$\left((u-u^{(0)}) - (u^{(m)}-u^{(0)}), v\right)_{\mathcal{A}} = 0, \quad \forall v \in \mathcal{K}_m.$$

The above $\mathcal{A}$-orthogonality gives

$$\begin{aligned}\left\|u-u^{(m)}\right\|_{\mathcal{A}} &= \min_{w\in\mathcal{K}_m}\left\|u-u^{(0)}-w\right\|_{\mathcal{A}} = \min_{q_{m-1}}\left\|u-u^{(0)}-q_{m-1}(\mathcal{A})r^{(0)}\right\|_{\mathcal{A}} \\ &= \min_{q_{m-1}}\left\|\left(I-q_{m-1}(\mathcal{A})\mathcal{A}\right)(u-u^{(0)})\right\|_{\mathcal{A}} = \min_{q_m(0)=1}\left\|q_m(\mathcal{A})(u-u^{(0)})\right\|_{\mathcal{A}}.\end{aligned}$$

Discussion on Convergence of CG Method

- Convergence of the Richardson method (or gradient descent method)

$$\left\|u-u^{(m)}\right\|_{\mathcal{A}} \leq \left(\frac{\kappa(\mathcal{A})-1}{\kappa(\mathcal{A})+1}\right)^{m} \left\|u-u^{(0)}\right\|_{\mathcal{A}}.$$

- Convergence of the conjugate gradient method

$$\left\|u-u^{(m)}\right\|_{\mathcal{A}} \leq 2\left(\frac{\sqrt{\kappa(\mathcal{A})}-1}{\sqrt{\kappa(\mathcal{A})}+1}\right)^{m} \left\|u-u^{(0)}\right\|_{\mathcal{A}}.$$

These classical results are only upper bounds. They are usually pessimistic!

- Q: How accurate are these estimates (upper bounds)? Distribution of eigenvalues!
- Reduce condition number: Meijerink, van der Vorst 1977 (IC preconditioner) ...
- **Preconditioners:** Diagonal, SGS, SSOR, Incomplete Factorizations, Sparse Approximate Inverse, Domain Decomposition, Geometric Multigrid, Algebraic Multigrid, ...
- **Problem dependency:** Preconditioning usually requires the coefficient matrix somehow

Effective Condition Number

- Convergence of the conjugate gradient method based on the effective condition number
- For more details, see Axelsson 2003

$$\frac{\|u-u^{(m)}\|_{\mathcal{A}}}{\|u-u^{(0)}\|_{\mathcal{A}}} \le 2C\left(\frac{\sqrt{\kappa_{\text{eff}}(\mathcal{A})}-1}{\sqrt{\kappa_{\text{eff}}(\mathcal{A})}+1}\right)^{m-m_0}, \quad m \ge m_0$$

- decomposition: $\sigma(A) = \sigma_{\text{bad}}(A) \cup \sigma_{\text{eff}}(A)$ with m_0 entries in $\sigma_{\text{bad}}(A)$
- effective condition number $\kappa_{\text{eff}}(A) := \max \sigma_{\text{eff}} / \min \sigma_{\text{eff}}$
- constant $C := \max_{\lambda \in \sigma_{\text{eff}}(A)} \prod_{\mu \in \sigma_{\text{bad}}(A)} \left|1 - \frac{\lambda}{\mu}\right|$
- $C < 1$ if σ_{bad} are isolated large eigenvalues; $C \le \left|\kappa(A)-1\right|^{m_0}$ in general
  - Van der Sluis-Van der Vorst 1986; Kaasschieter 1988

Stationary Iterative Method, Revisited

- Reminder: Convergence analysis of the linear stationary iterative methods (Lecture 4)

  Let $\mathcal{B}$ be a symmetric iterator for the SPD operator $\mathcal{A}$. An iterative method is convergent if

$$\rho := \rho(\mathcal{I} - \mathcal{B}\mathcal{A}) = \|\mathcal{I} - \mathcal{B}\mathcal{A}\|_{\mathcal{A}} < 1.$$

  The method is not only converging but also a contraction, i.e.,

$$\|u - u^{(m)}\|_{\mathcal{A}} \leq \rho^m \|u - u^{(0)}\|_{\mathcal{A}} \to 0 \quad \text{as } m \to +\infty.$$

- By definition, we have the following relation: $\Big((\mathcal{A} - 2\mathcal{A}\mathcal{B}\mathcal{A} + \mathcal{A}\mathcal{B}\mathcal{A}\mathcal{B}\mathcal{A})u, u\Big) \leq \rho^2 (u, u)_{\mathcal{A}}.$

Remark (Another equivalent condition)

If $\mathcal{A}$ and $\mathcal{B}$ are symmetric positive definite operators on a finite-dimensional space V, $\alpha > 0$ and $0 < \delta < 1$, then it is easy to verify the following two conditions are equivalent:

$$-\alpha(\mathcal{A}u, u) \leq (\mathcal{A}(\mathcal{I} - \mathcal{B}\mathcal{A})u, u) \leq \delta(\mathcal{A}u, u), \quad \forall u \in V \tag{16}$$

and

$$(1+\alpha)^{-1}(\mathcal{A}u, u) \leq (\mathcal{B}^{-1}u, u) \leq (1-\delta)^{-1}(\mathcal{A}u, u), \quad \forall u \in V. \tag{17}$$

Stationary Iterative Method as Preconditioner

By changing variable $v = \mathcal{A}^{1/2}u$, we obtain

$$\begin{aligned}\left((\mathcal{I} - \mathcal{A}^{1/2}\mathcal{B}\mathcal{A}^{1/2})^2 v, v\right) \le \rho^2(v,v) \quad &\Longrightarrow \quad \left|((\mathcal{I} - \mathcal{A}^{1/2}\mathcal{B}\mathcal{A}^{1/2})v, v)\right| \le \rho(v,v) \\ &\Longrightarrow \quad \left|((\mathcal{A} - \mathcal{A}\mathcal{B}\mathcal{A})u, u)\right| \le \rho(\mathcal{A}u, u), \quad \forall u \in V.\end{aligned}$$

Hence the condition number is uniformly bounded, i.e.,

$$\kappa(\mathcal{B}\mathcal{A}) \le \frac{1+\rho}{1-\rho}.$$

In fact, the above estimate can also be easily obtained from $\rho(\mathcal{I} - \mathcal{B}\mathcal{A}) = \rho < 1$. Indeed, the convergence rate of the preconditioned CG method is equal to (supposing $0 < \rho < 1$)

$$\delta_{\mathrm{CG}} := \frac{\sqrt{\kappa(\mathcal{B}\mathcal{A})} - 1}{\sqrt{\kappa(\mathcal{B}\mathcal{A})} + 1} \le \frac{\sqrt{\frac{1+\rho}{1-\rho}} - 1}{\sqrt{\frac{1+\rho}{1-\rho}} + 1} = \frac{1 - \sqrt{1-\rho^2}}{\rho} < \rho.$$

Preconditioned CG Method

- Q: How to understand solving the preconditioned system with CG?

$$\mathcal{B}\mathcal{A}u = \mathcal{B}f$$

- Define a new inner product: $(\mathcal{B}\mathcal{A}\cdot,\cdot)_{\mathcal{B}^{-1}}$ ⟹ SPD w.r.t. this inner product

Algorithm 3: Preconditioned conjugate gradient method

```
%% Given an initial guess u and a tolerance ε;
r ← f − Au,  p ← Br;
while ‖r‖ > ε
    α ← (Br, r)/(Ap, p);
    ũ ← u + αp;
    r̃ ← r − αAp;
    β ← (Br̃, r̃)/(Br, r);
    p̃ ← Br̃ + βp;
    Update:  u ← ũ,  r ← r̃,  p ← p̃;
end
```

Stopping Criteria

- Q: When can we stop the iterative procedure with confidence?

$$\mathcal{A}(u-u^{(m)}) = f - \mathcal{A}u^{(m)} = r^{(m)} \quad \Rightarrow \quad \|u-u^{(m)}\| \le \|\mathcal{A}^{-1}\|\|r^{(m)}\|$$

$$\|f\| \le \|\mathcal{A}\|\|u\| \quad \Rightarrow \quad \boxed{\frac{\|u-u^{(m)}\|}{\|u\|} \le \kappa(\mathcal{A})\frac{\|r^{(m)}\|}{\|f\|}.}$$

- If a good preconditioner is available, we can use a different norm.

$$\|e^{(m)}\|_{\mathcal{A}}^2 = \left(e^{(m)}, e^{(m)}\right)_{\mathcal{A}} = \left(r^{(m)}, e^{(m)}\right) \le \|r^{(m)}\|_{\mathcal{A}^{-1}} \|e^{(m)}\|_{\mathcal{A}}$$

residual-type a-post error analysis

$$\boxed{\|r^{(m)}\|_{\mathcal{A}^{-1}} \approx \|r^{(m)}\|_{\mathcal{B}}}$$

$$\left(r^{(m)}, r^{(m)}\right)_{\mathcal{B}} = \left(\mathcal{A}e^{(m)}, \mathcal{A}e^{(m)}\right)_{\mathcal{B}} = \left(\mathcal{A}\mathcal{B}\mathcal{A}e^{(m)}, e^{(m)}\right)$$

good preconditioner

Section 2. Generalized Minimal Residual Method

Krylov Matrices

- Cayley–Hamilton Theorem (Frobenius 1878): There exists a polynomial $q_{n-1}(\lambda) \in \mathcal{P}_{n-1}$, such that

$$A^{-1} = q_{n-1}(A)$$

characteristic polynomial

- The Krylov matrix can be defined as $K_n := \left[r, Ar, A^2r, \ldots, A^{n-1}r\right]$

- Get a similar transformation:

$$AK_n = \left[Ar, A^2r, \ldots, A^nr\right] = K_n \left[e_2, e_3, \ldots, e_n, K_n^{-1}A^nr\right] =: K_nC_n$$

$$\Rightarrow \quad C_n = K_n^{-1}AK_n$$

Upper Hessenberg

- Apply the QR factorization: $K_n = Q_nR_n$

$$R_n^{-1}Q_n^*AQ_nR_n = C_n$$

$$Q_n^*AQ_n = R_nC_nR_n^{-1} =: H_n$$

$$\Rightarrow \quad \boxed{Q_n^*AQ_n = H_n}$$

Krylov Subspace Methods

- The above Krylov matrix approach is NOT quite useful in iterative methods:
  - Stability: Nearly-singular! Ill conditioned! ← It's a power sequence
  - Efficiency: Expensive to compute the full QR factorization ← n is large
- The Krylov subspace

Nested Subspaces

$$\mathcal{K}_m(A, r) := \text{span}\{r, Ar, A^2r, \ldots, A^{m-1}r\}$$

The Krylov subspace methods: "Top Ten Algorithms of the Century", Dongarra and Sullivan, Computing in Science and Engineering, 2000

- Examples: CG, MinRes, GMRES, BiCGstab, FOM, GCR, ORTHOMIN, ... (see Y. Saad 2003)
- We will focus on the generalized minimum residual (GMRES) method:

$$\min_{e \in \mathcal{K}_m(A,r)} \|r - Ae\| \quad \Longrightarrow \quad \text{Line 4 of Algorithm 1}$$

Ref: V. Simoncini and D.B. Szyld. "Recent computational developments in Krylov subspace methods for linear systems." Numerical Linear Algebra with Applications 14.1, 1-59, 2007

Basic Ideas of GMRES

- Suppose that we have an **orthonormal basis** of the Krylov subspace

Saad and Schultz 1986

$$\mathcal{K}_m := \mathcal{K}_m(A, r) = \mathrm{span}\{q_1, q_2, \ldots, q_m\}$$

1 $\boxed{Q_m := [q_1, \ldots, q_m]}$

- Solve a least squares (LSQ) problem to find an "optimal" solution in the Krylov subspace

$$\text{Find } e_m \in \mathcal{K}_m \implies e_m = Q_m y, \quad y \in \mathbb{C}^m$$

$$\min_{e_m \in \mathcal{K}_m} \left\| r - A e_m \right\| \longrightarrow \min_{y \in \mathbb{C}^m} \left\| r - A Q_m y \right\| \longrightarrow \min_{y \in \mathbb{C}^m} \left\| r - Q_{m+1} \bar{H}_m \, y \right\|$$

2 $\boxed{AQ_m = Q_{m+1}\bar{H}_m}$

$$\longrightarrow \min_{y \in \mathbb{C}^m} \left\| Q_{m+1}^* r - \bar{H}_m \, y \right\| \longrightarrow \min_{y \in \mathbb{C}^m} \left\| \beta e_1 - \bar{H}_m \, y \right\|, \quad \beta := \|r\|$$

Multiply unitary matrix does not change 2-norm

3 $$q_1 := \frac{r}{\|r\|} \longrightarrow q_1^* r = \|r\|, \quad q_j^* r = 0, \quad j = 2, 3, \ldots$$

Only need to solve a small LSQ problem

Eigenvalue Equations

- Suppose that we have the Hessenberg decomposition of A:

$$A = QHQ^*, \quad Q : \text{orthogonal}, \quad H : \text{upper Hessenberg}$$

- We want part of this decomposition

$$A[q_1, \ldots, q_m, q_{m+1}, \ldots, q_n] = [q_1, \ldots, q_m, q_{m+1}, \ldots, q_n]H$$

$$= [q_1, \ldots, q_m, q_{m+1}, \ldots, q_n] \cdot \bar{H}_m$$

$$\bar{H}_m = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1m} & \cdots & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & \cdots & h_{2m} & \cdots & \cdots & h_{2n} \\ 0 & h_{32} & \ddots & \cdots & h_{3m} & \cdots & \cdots & h_{3n} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \cdots & \ddots & h_{m,m-1} & h_{mm} & \cdots & \vdots & \vdots \\ \vdots & \cdots & \cdots & 0 & h_{m+1,m} & \ddots & \vdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & h_{n,n-1} & h_{nn} \end{bmatrix}$$

2

Toward A Practical Iterative Procedure

- With the eigenvalue equation, we have

$$A[q_1, \ldots, q_m] = [q_1, \ldots, q_m, q_{m+1}]\, H(1 : m+1, 1 : m)$$

$$\boxed{AQ_m = Q_m H_m + h_{m+1,m} q_{m+1} e_m^T}$$

- How to develop an iterative procedure?

$$\boxed{AQ_m = Q_{m+1}\bar{H}_m, \quad A \in \mathbb{C}^{n\times n}, Q_m \in \mathbb{C}^{n\times m}, \bar{H}_m \in \mathbb{C}^{(m+1)\times m}} \quad (2)$$

$$\downarrow$$

$$Aq_m = h_{1m}q_1 + h_{2m}q_2 + \cdots + h_{mm}q_m + h_{m+1,m}q_{m+1}$$

$$\downarrow$$

$$q_{m+1} = \frac{Aq_m - \sum_{i=1}^{m} h_{im} q_i}{h_{m+1,m}}$$

We need to find a better iterative procedure to construct q-vectors!

Arnoldi Iteration

- Assume that we start from the residual vector

$$q_1 := \text{normalized initial residual} = \frac{r}{\|r\|}$$

3

- Gram-Schmidt orthogonalization

$$q_2 = \frac{Aq_1 - h_{11}q_1}{h_{21}} \quad \xrightarrow{q_1^* q_2 = 0} \quad \boxed{h_{11} = q_1^* A q_1}$$

$$w_2 := Aq_1 - h_{11}q_1$$

$$q_2 = \frac{w_2}{h_{21}}, \quad \boxed{h_{21} = \|w_2\|} \quad \longrightarrow \quad Q,\ H$$

Practical Implementation of Arnoldi

Algorithm 4: Arnoldi algorithm with Gram-Schmidt

```
1  %% Given an initial residual vector r ∈ ℝ^n;
2  for j = 1, 2, ..., m
3       h_{i,j} ← (A q_j, q_i),  i = 1 : j;
4       w_j ← A q_j − Σ_{i=1}^{j} h_{i,j} q_i;      ⟵ Standard Gram-Schmidt, not stable!
5       h_{j+1,j} ← ‖w_j‖;                          ➔ Modified Gram-Schmidt ➔ Reorthogonalization
6       if h_{j+1,j} == 0, break; %% lucky breakdown
7       q_{j+1} ← w_j / h_{j+1,j};
8  end
```

- Arnoldi algorithm breaks down at step j, if and only if the minimal polynomial of the vector q_1 (i.e. $p(A)q_1 = 0$) is of degree j
- See Proposition 6.6 in Y. Saad, "Iterative Methods for Sparse Linear Systems" (2nd Edition), 2003

Classical Gram-Schmidt Method

- Orthogonalization

$$\{a_1, a_2, \ldots, a_{j-1}\} \Longrightarrow \{q_1, q_2, \ldots, q_{j-1}\} \quad \text{Orthonormal}$$

$$\text{Add a new vector } a_j \Longrightarrow q_j \ ?$$

- Classical G-S:

$$w_j = a_j$$

$$\left.\begin{aligned} h_{ij} &= q_i^* a_j \\ w_j &= w_j - h_{ij} q_i \end{aligned}\right\} \ i = 1 : j-1 \Longrightarrow \left\{\begin{aligned} w_j &= a_j - \sum_{i=1}^{j-1} h_{ij} q_i \\ q_1^* w_j &= q_1^* a_j - h_{1j} = 0 \end{aligned}\right.$$

$$h_{jj} = \|w_j\|$$

$$q_j = \frac{w_j}{h_{jj}}$$

Modified Gram-Schmidt Method

- Orthogonalization

$$\{a_1, a_2, \ldots, a_{j-1}\} \Longrightarrow \{q_1, q_2, \ldots, q_{j-1}\} \quad \text{Orthonormal}$$

$$\text{Add a new vector } a_j \Longrightarrow q_j \ ?$$

- Modified G-S:

$$w_j = a_j$$

$$\left.\begin{aligned} h_{ij} &= q_i^* w_j \\ w_j &= w_j - h_{ij} q_i \end{aligned}\right\} \ i = 1 : j-1 \Longrightarrow \begin{cases} w_j = a_j - \sum_{i=1}^{j-1} h_{ij} q_i \\ q_1^* w_j = q_1^* a_j - h_{1j} = 0 \end{cases}$$

$$h_{jj} = \|w_j\|$$

$$q_j = \frac{w_j}{h_{jj}}$$

MGS-GMRES

Algorithm 5: Generalized minimum residual method

```
1  %% Given a nonsingular matrix A ∈ R^{n×n}, b ∈ R^n, and an initial guess x ∈ R^n;
2  r ← b − Ax, β ← ‖r‖, q_1 ← r/β;
3  for j = 1, 2, ..., m
4       w_j ← Aq_j;
5       h_ij ← (w_j, q_i), w_j ← w_j − h_ij q_i,   i = 1, 2, ..., j;
6       h_{j+1,j} ← ‖w_j‖;
7       if h_{j+1,j} == 0
8            m ← j, break;
9       end
10      q_{j+1} ← w_j/h_{j+1,j};
11 end
12 H̄_m ← {h_ij}_{1≤i≤m+1, 1≤j≤m};
13 y_m ← argmin_y ‖βe_1 − H̄_m y‖;
14 Update: x ← x + Q_m y_m;
```

Lucky breakdown: If and only if the iterative solution is exact!

- GMRES: Householder variant
- GMRES with restarting (stagnation)
- GMRES with deflated restarting
- GMRES with variable restarting

Stability of GMRES

Backward error analysis for GMRES with MGS in finite-precision arithmetic

$$\frac{\|b - Ax_m\|}{\|A\|\|x_m\| + \|b\|} \leq O(m^{2.5})\,\varepsilon$$

- Cannot store too many iterations for large linear systems!
- Cannot maintain orthogonality and numerical stability due to floating-point error!
- We have to restart the iteration ➔ GMRES(m)
- Forward error for direct methods: discussed in **Lecture 2, Summer 2022**
- Finite-precision arithmetic (floating-point calculation): discussed in **Lecture 5, Summer 2022**

Ref: Christopher C. Paige, Miroslav Rozložník, and Zdeněk Strakoš. “Modified Gram–Schmidt (MGS), least squares, and backward stability of MGS-GMRES”. SIAM Journal on Matrix Analysis and Applications, 2006; 28:264–284

Convergence of GMRES

$$r^{(k)} = f - Au^{(k)} = r^{(0)} - Aq_{k-1}(A)r^{(0)} =: p_k(A)r^{(0)}$$

- If A is diagonalizable, i.e. $A = Z\,\Lambda\,Z^{-1}$, then

$$\Rightarrow \quad \|r^{(k)}\| \le \max_{i=1,\dots,n} |p_k(\lambda_i)|\ \kappa(Z)\ \|r^{(0)}\|$$

- If A is normal, then Z is unitary ➔ $\|Z\| = \|Z^{-1}\| = 1$ ➔ Only need to analyze $p_k(\lambda_i)$

- Classical estimate [Elman 1982; Saad, Schultz 1986]: If $A = Z\Lambda Z^{-1}$ is diagonalizable and $\Lambda := \mathrm{diag}(\lambda_1, \dots, \lambda_n)$, then

$$\|r^{(k)}\| \le \|r^{(0)}\| \cdot \kappa(Z) \min_{p_k \in \mathcal{P}_k} \max_{i=1,\dots,n} \big|p_k(\lambda_i)\big|.$$

- If A is real and positive definite, then

$$\|r^{(k)}\| \le \|r^{(0)}\| \left(1 - \frac{a}{b}\right)^{\frac{k}{2}}, \quad a = \lambda_{\min}\left(\frac{A + A^T}{2}\right)^2, \quad b = \lambda_{\max}(A^T A).$$

Convergence Behavior of GMRES

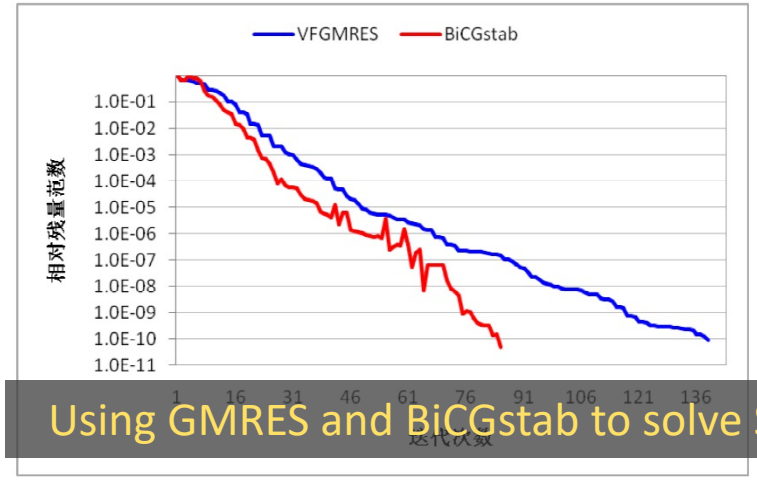
SIAM J. MATRIX ANAL. APPL.
Vol. 17, No. 3, pp. 465–469, July 1996

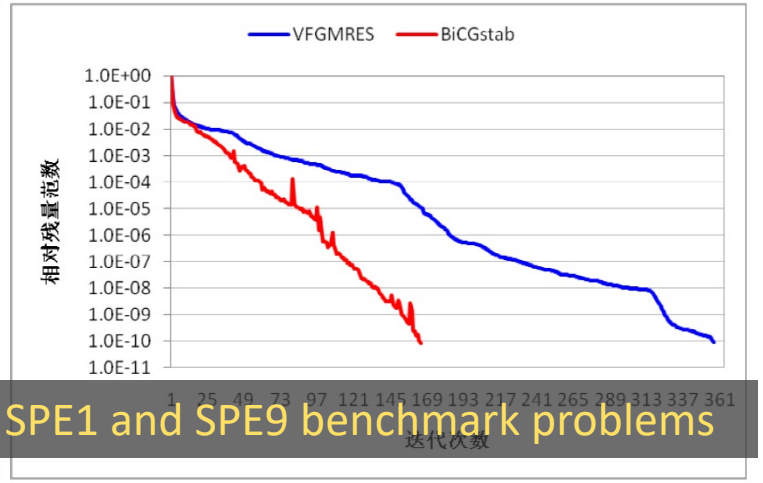
001

ANY NONINCREASING CONVERGENCE CURVE IS POSSIBLE FOR GMRES*

ANNE GREENBAUM†, VLASTIMIL PTÁK‡, AND ZDENĚK STRAKOŠ‡

Abstract. Given a nonincreasing positive sequence $f(0) \geq f(1) \geq \cdots \geq f(n-1) > 0$, it is shown that there exists an n by n matrix A and a vector r^0 with $\|r^0\| = f(0)$ such that $f(k) = \|r^k\|$, $k = 1, \ldots, n-1$, where r^k is the residual at step k of the GMRES algorithm applied to the linear system $Ax = b$, with initial residual $r^0 = b - Ax^0$. Moreover, the matrix A can be chosen to have any desired eigenvalues.





Using GMRES and BiCGstab to solve SPE1 and SPE9 benchmark problems

- Need good preconditioners
- If m gets too large, we cannot store everything; need to truncate or restart
- Restarts may kill convergence!
- Reuse the previous iterations after restarts? Do not forget everything when restarted
- Different preconditioners at each iteration?

Preconditioned GMRES Method

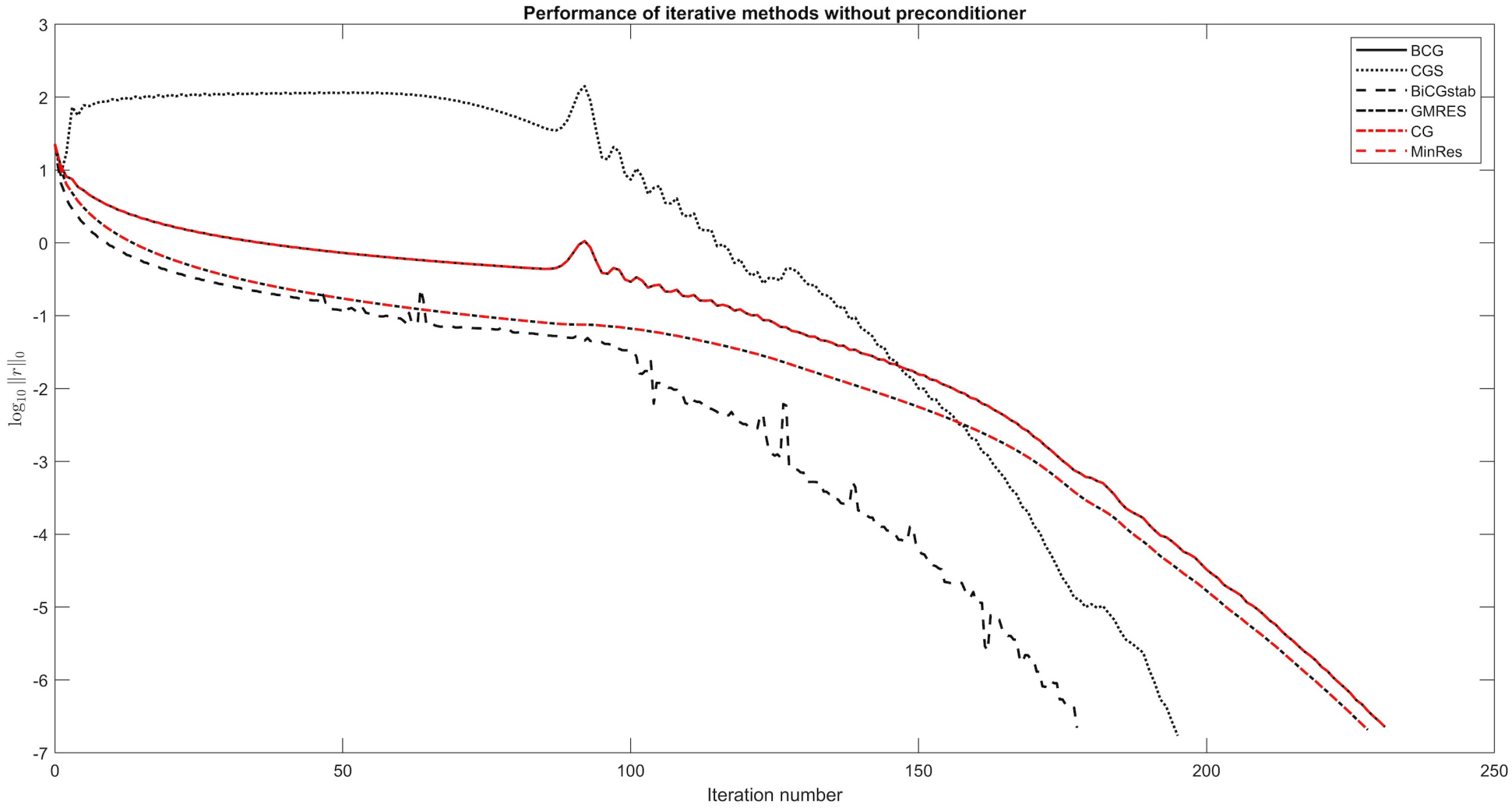
Algorithm 6: GMRES method with right preconditioner

1 %% Given a nonsingular matrix $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, and an initial guess $x \in \mathbb{R}^n$;
2 $r \leftarrow b - Ax$, $\beta \leftarrow \|r\|$, $q_1 \leftarrow r/\beta$;
3 for $j = 1, 2, \ldots, m$
4 $w_j \leftarrow AM^{-1}q_j$;
5 $h_{ij} \leftarrow (w_j, q_i)$, $w_j \leftarrow w_j - h_{ij}q_i$, $i = 1, 2, \ldots, j$;
6 $h_{j+1,j} \leftarrow \|w_j\|$;
7 if $h_{j+1,j} == 0$
8 $m \leftarrow j$, break;
9 end
10 $q_{j+1} \leftarrow w_j/h_{j+1,j}$;
11 end
12 $\bar{H}_m \leftarrow \{h_{ij}\}_{1\leqslant i\leqslant m+1, 1\leqslant j\leqslant m}$;
13 $y_m \leftarrow \mathrm{argmin}_y \|\beta e_1 - \bar{H}_m y\|$;
14 Update: $x \leftarrow x + M^{-1}Q_m y_m$;

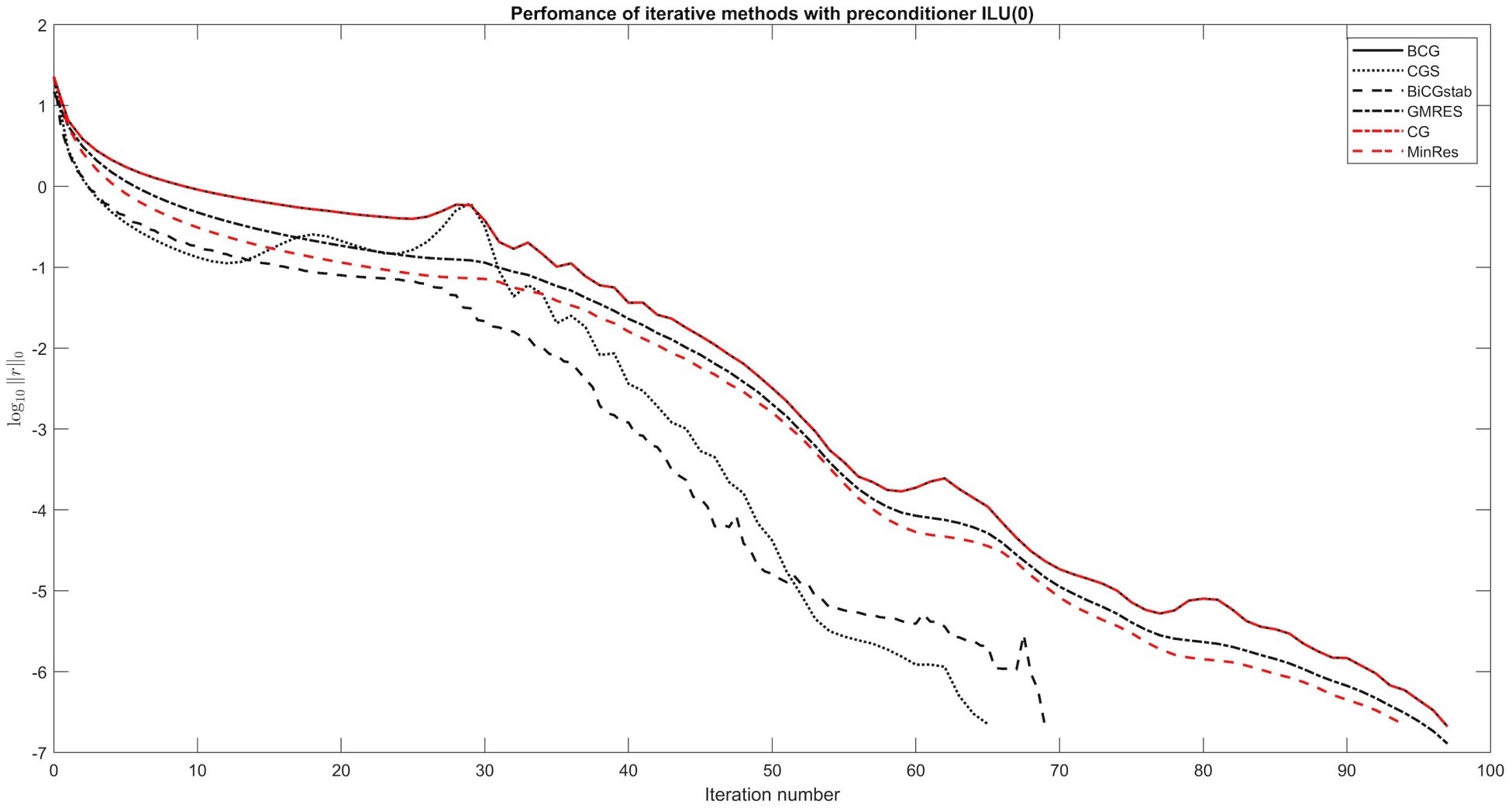
We do not save the preconditioned vectors because their sizes; instead, we just apply the preconditioner at Line 4 and Line 14. This way, need one more precondition step, but less memory is used!

$$\mathcal{K}_m(AM^{-1}, b) := \mathrm{span}\{b, AM^{-1}b, (AM^{-1})^2 b, \ldots, (AM^{-1})^{m-1}b\}$$

KSM for Poisson's Equation



Preconditioned KSM for Poisson's Equation



Flexible GMRES Method

Algorithm 7: Flexible GMRES method with right preconditioner

1 %% Given a nonsingular matrix $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, and an initial guess $x \in \mathbb{R}^n$;
2 $r \leftarrow b - Ax$, $\beta \leftarrow \|r\|$, $q_1 \leftarrow r/\beta$;
3 for $j = 1, 2, \ldots, m$
4 $z_j \leftarrow M_j^{-1} q_j$;
5 $w_j \leftarrow A z_j$;
6 $h_{ij} \leftarrow (w_j, q_i)$, $w_j \leftarrow w_j - h_{ij} q_i$, $i = 1, 2, \ldots, j$;
7 $h_{j+1,j} \leftarrow \|w_j\|$;
8 $q_{j+1} \leftarrow w_j / h_{j+1,j}$;
9 end
10 $\bar{H}_m \leftarrow \{h_{ij}\}_{1\leqslant i\leqslant m+1, 1\leqslant j\leqslant m}$;
11 $y_m \leftarrow \operatorname{argmin}_y \|\beta e_1 - \bar{H}_m y\|$;
12 Update: $x \leftarrow x + Z_m y_m$;

In order to allow different preconditioners at different steps, we have to adjust the PGMRES method and store the preconditioned vectors in FGMRES.

- FGMRES is very useful in practice, but oftentimes forgot
- But pay attention: Lucky breakdown no more!

Finding Eigenvalues using Krylov Methods

- Consider the eigenvalue problem

$$Ax = \lambda x$$

- Approximate the eigenvectors using the Krylov subspace

$$x = Q_k y_k + r \approx Q_k y_k$$

- Approximate eigenvalue problem in the Krylov subspace (Ritz values)

$$A\,Q_k y_k \approx \lambda\, Q_k y_k$$

$$H_k y_k = Q_k^* A Q_k y_k \approx \lambda y_k$$

$$H_k y = \mu y$$

- This gives a way to approximate few eigen-pairs of the original problem

Take Care of Small Eigenvalues

- Small eigenvalues can cause troubles for Krylov subspace methods
- Add approximate eigenvectors targeting the “smallest eigenvalues” to the Krylov subspace

$$\mathrm{span}\{r, Ar, A^2r, \ldots, A^{m-1}r, \; v_1, \ldots, v_k\}$$

- Q: How to obtain these eigenvectors?
- Approximate eigen information using Krylov subspaces (e.g. Paige, Parlett, van der Vorst 1995)

Good approximations of $\{v_1, \ldots, v_k\}$

$$H_m + h_{m+1,m}^2 H_m^{-T} e_m e_m^T \;\Rightarrow\; (\tilde{\theta}_i, \tilde{g}_i) \;\Rightarrow\; \tilde{y}_i := Q_m \tilde{g}_i$$

Harmonic Ritz values

Find smallest eigenpairs

$$\Rightarrow \mathrm{span}\{r, Ar, A^2r, \ldots, A^{m-1}r, \; \tilde{y}_1, \ldots, \tilde{y}_k\} \Rightarrow$$

GMRES-E (Morgan 1995)