AMS526: Numerical Analysis I
(Numerical Linear Algebra)
Lecture 23: GMRES and Other Krylov Subspace Methods; Preconditioning

Xiangmin Jiao

SUNY Stony Brook
Outline

1. Krylov Subspace Methods for Asymmetric Systems

2. Preconditioned Conjugate Gradient
Minimizing Residual

- CG only works for SPD matrices. It minimizes $\|x - x^*\|_A$ and $\phi(x) = \frac{1}{2}x^TAx - x^Tb$
- There have been many proposed extensions to nonsymmetric matrices, GMRES, BiCG, etc.
- GMRES (Generalized Minimal RESiduals) is one of most well known
- The basic idea of GMRES is to find $x_n \in \mathcal{K}_n$ that minimizes $\|r_n\| = \|b - Ax_n\|$
- This can be viewed as a least squares problem: Find a vector $c$ s.t. $\|AK_n c - b\|$ is minimized, where $K_n$ is the $m \times n$ Krylov matrix composed of basis vectors of $\mathcal{K}_n$, and

$$K_n = \begin{bmatrix}
b & Ab & \cdots & A^{n-1}b
\end{bmatrix}$$

- Orthogonal basis is often used, produced by Arnoldi iteration
Review: Arnoldi Iteration

- Let $Q_n = [q_1 \ | \ q_2 \ | \ \cdot \ | \ q_n]$ be $m \times n$ matrix with first $n$ columns of $Q$ and $\tilde{H}_n$ be $(n+1) \times n$ upper-left section of $H$
- Start by picking a random $q_1$ and then determine $q_2$ and $\tilde{H}_1$
- The $n$th columns of $AQ_n = Q_{n+1}\tilde{H}_n$ can be written as

$$Aq_n = h_{1n}q_1 + \cdots + h_{nn}q_n + h_{n+1,n}q_{n+1}$$

Algorithm: Arnoldi Iteration

given random nonzero $b$, let $q_1 = b/\|b\|$  
for $n = 1, 2, 3, \ldots$

$$v = Aq_n$$

for $j = 1$ to $n$

$$h_{jn} = q_j^*v$$

$$v = v - h_{jn}q_j$$

$$h_{n+1,n} = \|v\|$$

$$q_{n+1} = v/h_{n+1,n}$$
Minimal Residual with Orthogonal Basis

- Let $Q_n$ be Krylov matrix whose columns $q_1, q_2, \ldots$ span the successive Krylov subspaces.
- Instead of find $x_n = K_n c$, find $x_n = Q_n y$ which minimizes $\|AQ_n y - b\|$
- For Arnoldi iteration, we showed that $AQ_n = Q_{n+1} \tilde{H}_n$, so

\[
\|Q_{n+1} \tilde{H}_n y - b\| = \text{minimum}
\]

- Left multiplication by $Q_{n+1}^*$ does not change the norm, so

\[
\|\tilde{H}_n y - Q_{n+1}^* b\| = \text{minimum}
\]

- Finally, by construction, $Q_{n+1}^* b = \|b\| e_1$, so

\[
\|\tilde{H}_n y - \|b\| e_1\| = \text{minimum}.
\]
The GMRES Algorithm

Algorithm: GMRES

\[ q_1 = b / \| b \| \]

\textbf{for} \ n = 1, 2, 3, \ldots

\begin{itemize}
  \item Step \ n \ of \ Arnoldi \ iteration
  \item Find \ \( y \) \ to \ minimize \ \( \| \tilde{H}_n y - b e_1 \| = \| r_n \| \)
  \item \( x_n = Q_n y \)
\end{itemize}

\begin{itemize}
  \item The residual \( \| r_n \| \) \ does \ not \ need \ to \ be \ computed \ explicitly \ from \ \( x_n \)
\end{itemize}
The GMRES Algorithm

- Least squares problem has Hessenberg structure, solved with QR factorization of $\tilde{H}_n$
- If QR factorization of $\tilde{H}_n$ is constructed from scratch, then it costs $O(n^2)$ flops, due to Hessenberg structure
- However, QR factorization of $\tilde{H}_n$ can be updated from that of $\tilde{H}_{n-1}$, using Givens rotation within $O(n)$ work
- However, memory and cost grow with $n$.
- In practice, restart the algorithm by clearing accumulated data. This might stagnate the method.
GMRES and Polynomial Approximation

- GMRES can be interpreted as finding polynomial \( p_n \in P_n \) for \( n = 1, 2, 3, \ldots \) where

\[
P_n = \{ \text{polynomial } p \text{ of degree } \leq n \text{ with } p(0) = 1 \}
\]

such that \( \| p_n(A)b \| \) is minimized

- Note that \( r = b - AK_n c \), where

\[
AK_n c = (c_1 A + c_2 A^2 + \cdots + c_n A^n) b
\]

and \( r = (1 - c_1 A - c_2 A^2 - \cdots - c_n A^n) b \).

- In other words, \( p_n(z) = 1 - z(c_1 + c_2 z + \cdots + c_{n-1} z^{n-1}) \)

- Invariance of GMRES
  - Scale invariance: If we change \( A \rightarrow \sigma A \) and \( b \rightarrow \sigma b \), then \( r_n \rightarrow \sigma r_n \)
  - Invariance under unitary similarity transformations: If change \( A \rightarrow U A U^* \) for some unitary matrix \( U \) and \( b \rightarrow U b \), then \( r_n \rightarrow U^* r_n \)
Convergence of GMRES

- GMRES converges monotonically and it converges after at most \( m \) steps.
- Based on a polynomial analysis, diagonalizable \( A = V \Lambda V^{-1} \) converges as
  \[
  \frac{\| r_n \|}{\| b \|} \leq \kappa(V) \inf_{p_n \in P_n} \sup_{\lambda_i \in \Lambda(A)} |p_n(\lambda_i)|
  \]
- In other words, if \( A \) is not far from normal (i.e., eigenvectors are nearly orthogonal), and if properly normalized degree \( n \) polynomials can be found whose size on the spectrum \( \Lambda(A) \) decreases quickly with \( n \), then GMRES converges quickly.
Other Krylov Subspace Methods

- CG on the Normal Equations (CGN)
  - Solve $A^*Ax = A^*b$ using Conjugate Gradients
  - Poor convergence due to squared condition number (i.e., $\kappa(A^*A) = \kappa(A)^2$)
  - One advantage is that it applies least squares problems without modification

- BiConjugate Gradients (BCG/BiCG)
  - Makes residuals orthogonal to another Krylov subspace, based on $A^*$
  - It can be implemented with three-term recurrences, so memory requirements is smaller
  - Convergence sometimes comparable to GMRES, but unpredictable

- Conjugate Gradients Squared (CGS)
  - Avoids multiplication by $A^*$ in BCG, sometimes twice as fast convergence as BCG

- Quasi-Minimal Residuals (QMR) and Stabilized BiCG (Bi-CGSTAB)
  - Variants of BiCG with more regular convergence
MINRES: Minimum residual

- Solve \( Ax = b \) or \( (A - sI)x = b \). The matrix \( A - sI \) must be symmetric but it may be definite or indefinite or singular.
- Scalar \( s \) is a shifting parameter – it may be any number.
- The method is based on Lanczos tridiagonalization.
- MINRES is solving one of the least-squares problems minimize \( \|Ax - b\| \) or \( \|(A - sI)x - b\| \).

References
LSQR

- It solves $Ax = b$, or minimizes $\|Ax - b\|^2$ or minimize $\|Ax - b\|^2 + \lambda^2\|x\|^2$
- $A$ may be square or rectangular (over-determined or under-determined), and may have any rank
- The method is based on the Golub-Kahan bidiagonalization process. It is algebraically equivalent to applying MINRES to the normal equation $(A^TA + \lambda^2I)x = A^Tb$, but has better numerical properties, especially if $A$ is ill-conditioned
- LSQR reduces $\|r\|$ monotonically (where $r = b - Ax$ if $\lambda = 0$)
- References:
Outline

1. Krylov Subspace Methods for Asymmetric Systems

2. Preconditioned Conjugate Gradient
Preconditioning

- Motivation: Convergence of iterative methods heavily depends on eigenvalues or singular values of equation.
- Main idea of preconditioning is to introduce a nonsingular matrix $M$ such that $M^{-1}A$ has better properties than $A$. Thereafter, solve

$$M^{-1}Ax = M^{-1}b,$$

which has the same solution as $Ax = b$.

- Criteria of $M$
  - “Good” approximation of $A$, depending on iterative solvers
  - Ease of inversion

- Typically, a precondition $M$ is good if $M^{-1}A$ is not too far from normal and its eigenvalues are clustered.
Left, Right, and Hermitian Preconditioners

- Left preconditioner: Left multiply $M^{-1}$ and solve $M^{-1}Ax = M^{-1}b$
- Right preconditioner: Right multiply $M^{-1}$ and solve $AM^{-1}y = b$ with $x = M^{-1}y$
- However, if $A$ is Hermitian, $M^{-1}A$ or $AM^{-1}$ breaks symmetry
- How to resolve this problem?
Left, Right, and Hermitian Preconditioners

- **Left preconditioner:** Left multiply $M^{-1}$ and solve $M^{-1}Ax = M^{-1}b$
- **Right preconditioner:** Right multiply $M^{-1}$ and solve $AM^{-1}y = b$ with $x = M^{-1}y$
- However, if $A$ is Hermitian, $M^{-1}A$ or $AM^{-1}$ breaks symmetry
- How to resolve this problem?

Suppose $M$ is Hermitian positive definite, with $M = CC^*$ for some $C$, then $Ax = b$ is equivalent to

$$[C^{-1}AC^{-*}] (C^*x) = C^{-1}b,$$

where $C^{-1}AC^{-*}$ is Hermitian positive definite, and it is similar to $C^{-*}C^{-1}A = M^{-1}A$ and has same eigenvalues as $M^{-1}A$

- Example of $M = CC^*$ is Cholesky factorization $M = RR^*$, where $R$ is upper triangular
Preconditioned Conjugate Gradient

- When preconditioning a symmetric matrix, use SPD matrix $M$, and
  $M = RR^T$
- In practice, algorithm can be organized so that only $M^{-1}$ (instead of $R^{-1}$) appears

Algorithm: Preconditioned Conjugate Gradient Method

$$x_0 = 0, \ r_0 = b, \ p_0 = M^{-1}r_0, \ z_0 = p_0$$

for $n = 1, 2, 3, \ldots$

$$\alpha_n = (r_{n-1}^T z_{n-1})/(p_{n-1}^T A p_{n-1})$$

step length

$$x_n = x_{n-1} + \alpha_n p_{n-1}$$

approximate solution

$$r_n = r_{n-1} - \alpha_n A p_{n-1}$$

residual

$$z_n = M^{-1} r_n$$

preconditioning

$$\beta_n = (r_n^T z_n)/(r_{n-1}^T z_{n-1})$$

improvement this step

$$p_n = z_n + \beta_n p_{n-1}$$

search direction
Effective Preconditioners for CG

- SSOR Preconditioner
  - Simpler form: use matrix splitting of form $A = L + D + L^T$ and take
    
    $$M = (D + L)D^{-1}(D + L)^T$$
  
  - More generally, introduce SSOR relaxation parameter $\omega$, and take
    
    $$M = \frac{1}{2 - \omega}(\frac{1}{\omega}D + L)\left(\frac{1}{\omega}D\right)^{-1}\left(\frac{1}{\omega}D + L\right)^T.$$ 

    With optimal $\omega$, $\text{cond}(M^{-1}A) = O(\sqrt{\text{cond}(A)})$, but determining optimal $\omega$ is impractical

- Incomplete factorization
  - If $A = LL^T$ were used as preconditioner, then $\text{cond}(M^{-1}A) = 1$, but impractical
  
  - Instead, compute approximate factorization $A \approx \tilde{L}\tilde{L}^T$, which omit all fills or omit small fills and use $M = \tilde{L}\tilde{L}^T$ as preconditioner
Other Commonly Used Preconditioners

- **Jacobi preconditioning**: \( M = \text{diag}(A) \). Very simple and cheap, might improve certain problems but usually insufficient.

- **Block-Jacobi preconditioning**: Let \( M \) be composed of block-diagonal instead of diagonal.

- **Multigrid (coarse-grid approximations)**: For a PDE discretized on a grid, a preconditioner can be formed by transferring the solution to a coarser grid, solving a smaller problem, then transferring back. This is sometimes the most efficient approach if applicable.