Lecture 21
Overview of Iterative Methods

Songting Luo

Department of Mathematics
Iowa State University

MATH 562 Numerical Analysis II

Outline

Outline

Direct v.s. Iterative Methods

- Direct methods, or noniterative methods, compute the exact solution after a finite number of steps (in exact arithmetic)
  - Example: Gaussian elimination, QR factorization, LU, LDL, Cholsky
- Iterative methods produce a sequence of approximations $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots$ that hopefully converge to the true solution
  - Example: Jacobi, Gauss-Seidel, Conjugate Gradient (CG), GMRES, BiCG, etc.
- What have we learned before?
- Why use iterative methods (instead of direct methods)?
  - may be faster than direct methods
  - produce useful intermediate results
  - handle sparse matrices more easily (needs only matrix-vector product)
  - often are easier to implement on parallel computers
- Future?

Two Classes of Iterative Methods

- Stationary methods find a splitting $\mathbf{A} = \mathbf{M} - \mathbf{K}$ and iterates

$$\mathbf{x}^{(k+1)} = \mathbf{M}^{-1}(\mathbf{K}\mathbf{x}^{(k)} + \mathbf{b})$$

  - Examples: Jacobi (for linear systems, not the Jacobi iterations for eigenvalues), Gauss-Seidel, Successive Over-Relaxation (SOR) etc.

- Krylov subspace methods find optimal solution in Krylov subspace $\{\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \cdots, \mathbf{A}^{k-1}\mathbf{b}\}$
  - Build subspace successively
  - Example: Conjugate Gradient (CG), Generalized Minimum Residual (GMRES), BiCG, etc.
  - We will focus on Krylov subspace methods

Krylov Subspace Methods

- Given $\mathbf{A}$ and $\mathbf{b}$, Krylov subspace

$$\{\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}, \ldots, \mathbf{A}^{k-1}\mathbf{b}\}$$

	linear systems	eigenvalue problems
Hermitian	CG	Lanczos
Nonhermitian	GMRES, BiCG, etc.	Arnoldi

Krylov Subspace Algorithms

- Create a sequence of Krylov subspaces for $\mathbf{Ax} = \mathbf{b}$

$$\mathcal{K}_n = \{\mathbf{b}, \mathbf{Ab}, \ldots, \mathbf{A}^{n-1}\mathbf{b}\}$$

  and find an approximate (hopefully optimal) solutions $\mathbf{x}_n \in \mathcal{K}_n$
- Only matrix-vector products involved
- Some criterions of "optimal" at each iteration:
  - Residual $\mathbf{r}_n = \mathbf{b} - \mathbf{Ax}_n$ is orthogonal to $\mathcal{K}_n$ (**Conjugate Gradients**)
  - Residual $\mathbf{r}_n$ has minimum norm for $\mathbf{x}_n \in \mathcal{K}_n$ (**GMRES** and **MINRES**)
  - $\mathbf{r}_n$ is orthogonal to a different space $\mathcal{K}_n(\mathbf{A}^T)$ (**BiConjugate Gradients**)
  - The error $\mathbf{e}_n$ has minimum norm (**SYMMLQ**).

Arnoldi Iteration

- The Arnoldi iteration reduces a general, nonsymmetric matrix $\mathbf{A}$ to Hessenberg form by similarity transformation $\mathbf{A} = \mathbf{QHQ}^*$
- Analogous to Gram-Schmidt-style iteration instead of Householder reflections
- Let $\mathbf{Q}_n = [\mathbf{q}_1|\mathbf{q}_2|\cdots|\mathbf{q}_n]$ be $m \times n$ matrix with first n columns of $\mathbf{Q}$ and $\tilde{\mathbf{H}}_n$ be $(n+1) \times n$ upper-left section of $\mathbf{H}$.
- Consider first n columns of $\mathbf{AQ} = \mathbf{QH}$, or $\mathbf{AQ}_n = \mathbf{Q}_{n+1}\tilde{\mathbf{H}}_n$

$$[\mathbf{A}]\,[\mathbf{q}_1|\cdots|\mathbf{q}_n] = \left[\mathbf{q}_1|\cdots|\mathbf{q}_{n+1}\right] \begin{bmatrix} h_{11} & \cdots & h_{1n} \\ h_{21} & & \\ & \ddots & \vdots \\ & & h_{n+1,n} \end{bmatrix}$$

Arnoldi Algorithm

- Start by picking a random $\mathbf{q}_1$ and then determine $\mathbf{q}_2$ and $\tilde{\mathbf{H}}_1$
- The nth columns of $\mathbf{A}\mathbf{Q}_n = \mathbf{Q}_{n+1}\tilde{\mathbf{H}}_n$ can be written as

$$\mathbf{A}\mathbf{q}_n = h_{1n}\mathbf{q}_1 + \cdots + h_{nn}\mathbf{q}_n + h_{n+1,n}\mathbf{q}_{n+1}$$

Algorithm: Arnoldi Iteration

given random nonzero $\mathbf{b}$, let $\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|$
for $n = 1$ to $1, 2, 3, \ldots$
    $\mathbf{v} = \mathbf{A}\mathbf{q}_n$
    for $j = 1$ to n
        $h_{jn} = \mathbf{q}_j^* \mathbf{v}$
        $\mathbf{v} = \mathbf{v} - h_{jn}\mathbf{q}_j$
    $h_{n+1,n} = \|\mathbf{v}\|$
    $\mathbf{q}_{n+1} = \mathbf{v}/h_{n+1,n}$

- A version of Gram-Schmidt, tailored to Krylov subspaces.

QR Factorization of Krylov Matrix

- The vector $\mathbf{q}_j$ from Arnoldi are orthonormal bases of successive Krylov subspaces

$$\mathcal{K}_n = \langle \mathbf{b}, \mathbf{Ab}, \ldots, \mathbf{A}^{n-1}\mathbf{b} \rangle = \langle \mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n \rangle \subseteq \mathbb{C}^m$$

- $\mathbf{Q}_n$ is reduced QR factorization $\mathbf{K}_n = \mathbf{Q}_n \mathbf{R}_n$ of Krylov matrix

$$\mathbf{K}_n = \left[\mathbf{b} | \mathbf{Ab} | \cdots | \mathbf{A}^{n-1}\mathbf{b} \right]$$

- The projection of $\mathbf{A}$ onto this space gives $n \times n$ Hessenberg matrix $\mathbf{H}_n = \mathbf{Q}_n^* \mathbf{A} \mathbf{Q}_n = \tilde{\mathbf{H}}_{1:n,1:n}$
- Eigenvalues of $\mathbf{H}_n$ (known as Ritz values) produce good approximations of those of $\mathbf{A}$

Lanczos Iteration for Symmetric Matrices

- For symmetric $\mathbf{A}$, $\tilde{\mathbf{H}}_n$ $\mathbf{H}_n$ are tridiagonal, denoted by $\tilde{\mathbf{T}}_n$ and $\mathbf{T}_n$, respectively. $\mathbf{AQ}_n = \mathbf{Q}_{n+1}\tilde{\mathbf{H}}_n$ can be written as three-term recurrence
  $\mathbf{Aq}_n = \beta_{n-1}\mathbf{q}_{n-1} + \alpha_n\mathbf{q}_n + \beta_n\mathbf{q}_{n+1}$
  where α_i are diagonal entries and β_i are sub-diagonal entries of $\tilde{\mathbf{T}}_n$

Algorithm: Lanczos Iteration

$\beta_0,\ \mathbf{q}_0 = \mathbf{0}$
given random $\mathbf{b}$, let $\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|$
for $n = 1$ to $1, 2, 3, \ldots$
$\quad \mathbf{v} = \mathbf{Aq}_n$
$\quad \alpha_n = \mathbf{q}_n\mathbf{v}$
$\quad \mathbf{v} = \mathbf{v} - \beta_{n-1}\mathbf{q}_{n-1} - \alpha_n\mathbf{q}_n$
$\quad \beta_n = \|\mathbf{v}\|$
$\quad \mathbf{q}_{n+1} = \mathbf{v}/\beta_n$

- Eigenvalues of $\mathbf{T}_n$ (known as Ritz values) converge to eigenvalues of $\mathbf{A}$, and extreme eigenvalues converge faster

Properties of Arnoldi and Lanczos Iterations

- Eigenvalues of $\mathbf{H}_n$ (or $\mathbf{T}_n$ in Lanczos iterations) are called Ritz values.
- When $m = n$, Ritz values are eigenvalues.
- Even for $n \ll m$, Ritz values are often accurate approximations to eigenvalues of $\mathbf{A}$!
- For symmetric matrices with evenly spaced eigenvalues, Ritz values tend to first convert to extreme eigenvalue.
- With rounding errors, Lanczos iteration can suffer from loss of orthogonality and can in turn lead to spurious "ghost" eigenvalues.