EECS 275 Matrix Computation

Ming-Hsuan Yang

Electrical Engineering and Computer Science
University of California at Merced
Merced, CA 95344
http://faculty.ucmerced.edu/mhyang

Lecture 21

Overview

- Conjugate gradient
- Convergence rate of conjugate gradient
- Preconditioning

Reading

- Chapter 39-40 of *Numerical Linear Algebra* by Llyod Trefethen and David Bau
- Chapter 10 of *Matrix Computations* by Gene Golub and Charles Van Loan
- "An Introduction to Conjugate Gradient Method Without the Agonizing Pain" by Jonathan Shewchuk

Optimality of conjugate gradients

Theorem

Let the conjugate gradient iteration be applied to a symmetric positive definitive matrix problem $A\mathbf{x} = \mathbf{b}$. *If the iteration has not already converged (i.e.,* $\mathbf{r}_{n-1} \neq 0$*), then* $\mathbf{x}_n$ *is the unique point in* $\mathcal{K}_n$ *that minimizes* $\|\mathbf{e}_n\|_A$. *The convergence is monotonic*

$$\|\mathbf{e}_n\|_A \leq \|\mathbf{e}_{n-1}\|_A \tag{1}$$

and $\mathbf{e}_n = 0$ *is achieved for some* $n \leq m$

- From previous theorem, we know that $\mathbf{x}_n$ belongs to $\mathcal{K}_n$
- Consider an arbitrary point $\mathbf{x} = \mathbf{x}_n - \Delta\mathbf{x} \in \mathcal{K}_n$ with error $\mathbf{e} = \mathbf{x}_* - \mathbf{x} = \mathbf{e}_n + \Delta\mathbf{x}$, and compute

$$\begin{aligned} \|\mathbf{e}\|_A^2 &= (\mathbf{e}_n + \Delta\mathbf{x})^\top A(\mathbf{e}_n + \Delta\mathbf{x}) \\ &= \mathbf{e}_n^\top A\mathbf{e}_n + (\Delta\mathbf{x})^\top A(\Delta\mathbf{x}) + 2\mathbf{e}_n^\top A(\Delta\mathbf{x}) \end{aligned}$$

- The last term $2\mathbf{e}_n^\top A\Delta\mathbf{x} = 2\mathbf{r}_n^\top(\Delta\mathbf{x})$, an inner product of $\mathbf{r}_n$ with a vector in $\mathcal{K}_n$, is zero (using previous theorem)

Optimality of conjugate gradients (cont'd)

- An inner product of $\mathbf{r}_n$ and a vector in $\mathcal{K}_n$ is zero
- A crucial property that makes CG powerful
- It implies that

$$\|\mathbf{e}\|_A^2 = \mathbf{e}_n^\top A\mathbf{e}_n + (\Delta\mathbf{x})^\top A(\Delta\mathbf{x})$$

- Only the second term depends on $\Delta\mathbf{x}$ and since A is positive definite, the first term is larger or equal to 0
- The second term is 0 if and only if $\Delta\mathbf{x} = \mathbf{0}$, i.e., $\mathbf{x}_n = \mathbf{x}$
- Thus $\|\mathbf{e}\|_A$ is minimal if and only if $\mathbf{x}_n = \mathbf{x}$ as claimed
- The monotonicity property is a consequence of the inclusion $\mathcal{K}_n \subseteq \mathcal{K}_{n+1}$, and since $\mathcal{K}_n$ is a subset of $\mathbb{R}^m$ of dimension n as long as convergence has not yet been achieved, convergence must be achieved in at most m steps
- That is, each step of conjugate direction cuts down the error term component by component

Optimality of conjugate gradients (cont'd)

- The guarantee that the CG iteration converges in at most m steps is void in floating point arithmetic
- For arbitrary matrices A on a real computer, no decisive reduction in $\|\mathbf{e}_n\|_A$ will necessarily be observed at all when $n = m$
- In practice, however, CG is used not for arbitrary matrices but for matrices whose spectra are well behaved (partially due to preconditioning) that convergence to a desired accuracy is achieved for $n \ll m$
- The theoretical exact convergence at $n = m$ has no relevance to this use of the CG iteration in scientific computing

Conjugate gradients as an optimization algorithm

- The CG iteration has a certain optimality property: it minimizes $\|\mathbf{e}_n\|_A$ at step n over all vectors $\mathbf{x} \in \mathcal{K}_n$
- A standard form for minimizing a nonlinear function of $\mathbf{x} \in \mathbb{R}^m$
- At the heart of the iteration is the formula

$$\mathbf{x}_n = \mathbf{x}_{n-1} + \alpha_n \mathbf{p}_{n-1}$$

- A familiar equation in optimization, in which a current approximation $\mathbf{x}_{n-1}$ is updated to a new approximation $\mathbf{x}_n$ by moving a distance α_n (the step length) in the direction $\mathbf{p}_{n-1}$ (the search direction)
- By a succession of such steps, the CG iteration attempts to find a minimum of a nonlinear equation
- Which function to minimize?

Conjugate gradients as an optimization algorithm (cont'd)

- Cannot use $\|\mathbf{e}\|_A$ or $\|\mathbf{e}\|_A^2$ as neither can be evaluated without knowing $\mathbf{x}_*$
- On the other hand, given A and $\mathbf{b}$ and $\mathbf{x} \in \mathbb{R}^m$, the quantity

$$\phi(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top A\mathbf{x} - \mathbf{x}^\top \mathbf{b}$$

  can certainly be evaluated as

$$\begin{aligned} \|\mathbf{e}_n\|_A^2 &= \mathbf{e}_n^\top A\mathbf{e}_n = (\mathbf{x}_* - \mathbf{x}_n)^\top A(\mathbf{x}_* - \mathbf{x}_n) \\ &= \mathbf{x}_n^\top A\mathbf{x}_n - 2\mathbf{x}_n^\top A\mathbf{x}_* + \mathbf{x}_*^\top A\mathbf{x}_* \\ &= \mathbf{x}_n^\top A\mathbf{x}_n - 2\mathbf{x}_n^\top \mathbf{b} + \mathbf{x}_*^\top \mathbf{b} \\ &= 2\phi(\mathbf{x}_n) + \textit{constant} \end{aligned}$$

- Like $\|\mathbf{e}\|_A^2$, it must achieve its minimum uniquely at $\mathbf{x} = \mathbf{x}_*$

Conjugate gradients as an optimization algorithm (cont'd)

- The CG iteration can be interpreted as an iterative process for minimizing the quadratic function $\phi(\mathbf{x})$ of $\mathbf{x} \in \mathbb{R}^m$
- At each step, an iterate $\mathbf{x}_n = \mathbf{x}_{n-1} + \alpha_n \mathbf{p}_{n-1}$ is computed that minimizes $\phi(\mathbf{x})$ over all $\mathbf{x}$ in the one dimensional space $\mathbf{x}_{n-1} + \langle \mathbf{p}_{n-1} \rangle$
- It can be readily confirmed that the formula

$$\alpha_n = \frac{\mathbf{r}_{n-1}^\top \mathbf{r}_{n-1}}{\mathbf{p}_{n-1}^\top A \mathbf{p}_{n-1}}$$

  ensures α_n is optimal in the sense among all step lengths α
- What makes the CG iteration remarkable is the choice of the search direction $\mathbf{p}_{n-1}$, which has the special property that minimizing $\phi(\mathbf{x})$ over $\mathbf{x}_{n-1} + \langle \mathbf{p}_{n-1} \rangle$ actually minimizes it over all of $\mathcal{K}_n$

Analogy between CG iteration and Lanczos iteration

- A close analogy between CG iteration for solving $A\mathbf{x} = \mathbf{b}$ and the Lanczos iteration for finding eigenvalues
- The eigenvalues of A are the stationary values for $\mathbf{x} \in \mathbb{R}^m$ of the Rayleigh quotient $r(\mathbf{x}) = \frac{\mathbf{x}^\top A\mathbf{x}}{\mathbf{x}^\top \mathbf{x}}$
- The eigenvalue estimates (Ritz values) associated with step n of the Lanczos iteration are the stationary values of the same function $r(\mathbf{x})$ if $\mathbf{x}$ is restricted to the Krylov subspace $\mathcal{K}_n$
- Perfect parallel of what we have shown that the solution $\mathbf{x}_*$ of $A\mathbf{x} = \mathbf{b}$ is the minimal point in $\mathbb{R}^m$ of the scalar function $\phi(\mathbf{x})$, and the CG iterate $\mathbf{x}_n$ is the minimal point of the same function $\phi(\mathbf{x})$ if $\mathbf{x}$ is restricted to $\mathcal{K}_n$

Conjugate gradients and polynomial approximation

- Connection between Krylov subspace iteration and polynomials of matrices
- The Arnoldi and Lanczos iterations solve the Arnoldi/Lanczos approximation problem
  Find $p^n \in P^n$ such that
  $$\|p^n(A)\mathbf{b}\| = \text{minimum}$$
- The GMRES iteration solves the GMRES approximation problem
  Find $p_n \in P_n$ such that
  $$\|p_n(A)\mathbf{b}\| = \text{minimum}$$
- For CG, the appropriate approximation problem involves the A-norm of the error
  Find $p_n \in P_n$ such that
  $$\|p_n(A)\mathbf{e}_0\|_A = \text{minimum}$$

  where $\mathbf{e}_0$ denotes the initial error $\mathbf{e}_0 = \mathbf{x}_* - \mathbf{x}_0 = \mathbf{x}_*$, and P_n is again defined as GMRES (i.e., the set of polynomials p of degree $\leq n$ with $p(0) = 1$)

CG and polynomial approximation

Theorem

If the CG iteration has not already converged before step n (i.e., $\mathbf{r}_{n-1} \neq \mathbf{0}$), then $\|p_n(A)\mathbf{e}_0\|_A$ has a unique solution $p_n \in P_n$, and the iterate $\mathbf{x}_n$ has error $\mathbf{e}_n = p_n(A)\mathbf{e}_0$ for this same polynomial p_n. Consequently, we have

$$\frac{\|\mathbf{e}_n\|_A}{\|\mathbf{e}_0\|_A} = \inf_{p \in P_n} \frac{\|p(A)\mathbf{e}_0\|_A}{\|\mathbf{e}_0\|_A} \leq \inf_{p \in P_n} \max_{\lambda \in \Lambda(A)} |p(\lambda)| \tag{2}$$

where $\Lambda(A)$ denotes the spectrum of A

- From theorem in the last lecture, it follows that $\mathbf{e}_n = p(A)\mathbf{e}_0$ for some $p \in P_n$
- The equality is a consequence of (2) and monotonic convergence (1)

CG and polynomial approximation (cont'd)

- As for the inequality, $\mathbf{e}_0 = \sum_{j=1}^{m} a_j \mathbf{v}_j$ is an expansion of $\mathbf{e}_0$ in orthonormal eigenvectors of A, then we have $p(A)\mathbf{e}_0 = \sum_{j=1}^{m} a_j p(\lambda_j)\mathbf{v}_j$ and thus

$$\|\mathbf{e}_0\|_A^2 = \sum_{j=1}^{m} a_j^2 \lambda_j, \quad \|p(A)\mathbf{e}_0\|_A^2 = \sum_{j=1}^{m} a_j^2 \lambda_j (p(\lambda_j))^2$$

  These identities imply $\|p(A)\mathbf{e}_0\|_A^2 / \|\mathbf{e}_0\|_A^2 \leq \max_{\lambda \in \Lambda(A)} |p(\lambda)|^2$, which implies the inequality
- The rate of convergence of the CG iteration is determined by the location of the spectrum of A
- A good spectrum is one on which polynomials $p_n \in P_n$ can be very small, with size decreasing rapidly with n
- Roughly speaking, this may happen for either or both of two reasons: the eigenvalues may be grouped in small clusters, or they may lie well separated in a relative sense from the origin
- The two best known corollaries address these two ideas in their extreme forms

Rate of CG convergence

- First, we suppose that the eigenvalues are perfectly clustered but assume nothing about the locations of these clusters

Theorem

If A has only n distinct eigenvalues, then the CG iteration converges in at most n steps

- This is a corollary of (2), since a polynomial $p(\mathbf{x}) = \prod_{j=1}^{n}(1 - \mathbf{x}/\lambda_j) \in P_n$ exists that is zero at any specified set of n points $\{\lambda_j\}$
- At the other extreme, suppose we know nothing about any clustering of the eigenvalues but only that their distances from the origin vary by at most a factor $\kappa \geq 1$
- In other words, suppose we know only the 2-norm condition number $\kappa = \lambda_{max}/\lambda_{min}$, where λ_{max} and λ_{min} are the extreme eigenvalues of A

Rate of CG convergence (cont'd)

Theorem

Let the CG iteration be applied to a symmetric positive definite matrix problem $A\mathbf{x} = \mathbf{b}$, where A has 2-norm condition number κ. Then the A-norm of the errors satisfy

$$\frac{\|\mathbf{e}_n\|_A}{\|\mathbf{e}_0\|_A} \leq 2 \Big/ \left[\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1} \right)^n + \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1} \right)^{-n} \right] \leq 2 \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \right)^n$$

- See text for proof using Chebyshev polynomials
- Since

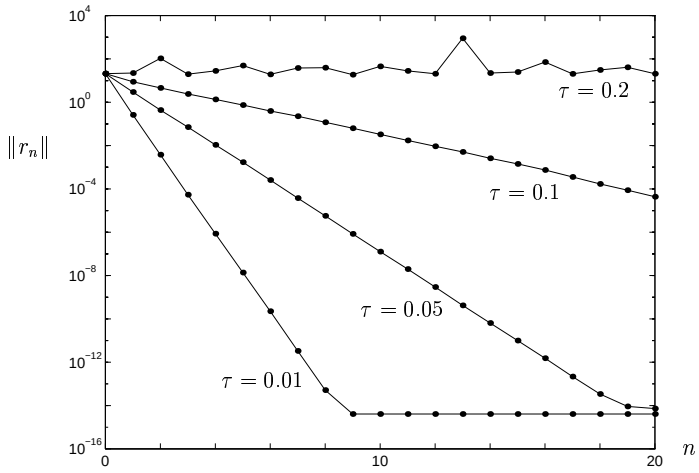
$$\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \sim 1 - \frac{2}{\sqrt{\kappa}}$$

  as $\kappa \to \infty$, it implies that if κ is large but not too large, convergence to a specified tolerance can be expected in $O(\sqrt{\kappa})$ iterations
- An upper bound, and convergence may be faster for special right hand sides or if the spectrum is clustered

Example: CG convergence

- Consider a 500×500 sparse matrix A where we have 1's on the diagonal and a random number from the uniform distribution on $[-1, 1]$ at each off-diagonal position (maintaining the symmetry $A = A^\top$)
- Then we replace each off-diagonal entry with $|A_{ij}| > \tau$ by zero, where τ is a parameter
- For τ close to zero, the result is a well-conditioned positive definite matrix whose density of nonzero entries is approximately τ
- As τ increases, both the condition number and the sparsity deteriorate

Example: CG convergence (cont'd)



- For $\tau = 0.01$, A has 3,092 nonzero entries and $\kappa \approx 1.06$, the CG convergence takes place in 9 steps
- For $\tau = 0.05$, A has 13,062 nonzero entries with $\kappa \approx 1.83$, and convergence takes place in 19 steps
- For $\tau = 0.1$, A has 25,526 nonzero entries with $\kappa \approx 10.3$ and the process converges in 20 steps
- For $\tau = 0.2$ with 50,834 nonzero entries, there is no convergence at all
- For this example, the CG beats Cholesky factorization by a factor of about 700 in terms of operation counts

Preconditioning

- The convergence of a matrix iteration depends on the properties of the matrix - the eigenvalues, the singular values, or sometimes other information
- In many cases, the problem of interest can be transformed so that the properties of the matrix are improved drastically
- The process of preconditioning is essential to most successful applications of iterative methods

Preconditioning for $A\mathbf{x} = \mathbf{b}$

- Suppose we want to solve $m \times m$ nonsingular system

$$A\mathbf{x} = \mathbf{b} \tag{3}$$

- For any nonsingular $m \times m$ matrix M, the system

$$M^{-1}A\mathbf{x} = M^{-1}\mathbf{b} \tag{4}$$

  has the same solution
- If we solve the (4) iteratively, however, the convergence will depend on the properties of $M^{-1}A$ instead of A
- If this preconditioner M is well chosen, (4) may be solved much more rapidly than (3)
- For this idea to be useful, it must be possible to compute $M^{-1}A$ efficiently
- As usual in numerical linear algebra, this does not mean an explicit construction of the inverse M^{-1}, but the solution of system of equations in this form

$$M\mathbf{y} = \mathbf{c} \tag{5}$$

Preconditioning for $A\mathbf{x} = \mathbf{b}$ (cont'd)

- Two extreme cases:
  - If $M = A$, then (5) is the same as (3), and nothing has been gained
  - If $M = I$, then (4) is the same as (3), and then it is a trivial solution
- Between these two extremes lie the useful preconditioners,
  - structured enough (5) can be solved quickly
  - but close enough to A in some sense that an iteration for (4) converges more quickly than an iteration for (3)
- What does it mean for M to be "close enough" to A?
- If the eigenvalues of $M^{-1}A$ are close to 1 and $\|M^{-1}A - I\|_2$ is small, then any of the iterations we have discussed can be expected to converge quickly
- However, preconditioners that do not satisfy such a strong condition may also perform well
- A simple rule of thumb: *preconditioner M is good if $M^{-1}A$ is not too far from normal and its eigenvalues are clustered*

Left, right and Hermitian preconditioners

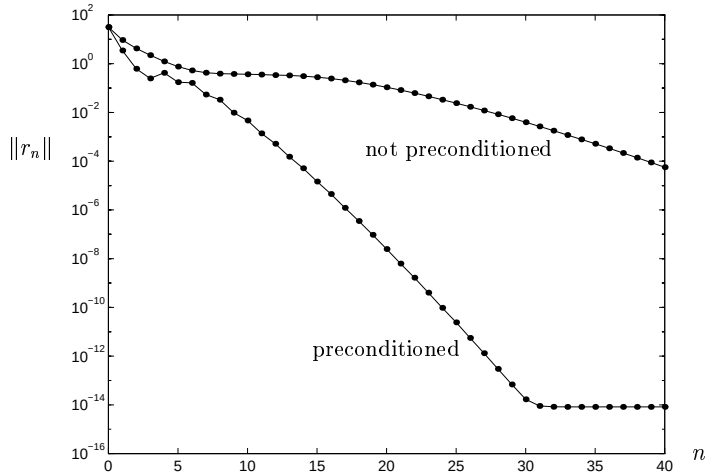
- What we have described may be more precisely terms as left preconditioner
- Another idea is to transform $A\mathbf{x} = \mathbf{b}$ into $AM^{-1}\mathbf{y} = \mathbf{b}$ with $\mathbf{x} = M^{-1}\mathbf{y}$ in which case M is called a right preconditioner
- If A is Hermitian positive definite, then it is usual to preserve this property in preconditioning
- Suppose M is also Hermitian positive definite, with $M = CC^*$ for some C, then (3) is equivalent to

$$[C^{-1}AC^{-*}]C^*\mathbf{x} = C^{-1}\mathbf{b}$$

- The matrix in brackets is Hermitian positive definite, so this equation can be solved by conjugate gradient or related iterations
- At the same time, since $C^{-1}AC^{-*}$ is similar to $C^{-*}C^{-1}A = M^{-1}A$, it is enough to examine the eigenvalues of the non-Hermitian matrix $M^{-1}A$ to investigate convergence

Example: Preconditioning convergence

- Consider a 1000×1000 symmetric matrix A whose entries are all zero except for $A_{ij} = 0.5 + \sqrt{i}$ on the diagonal, $A_{ij} = 1$ on the sub-and super-diagonals, and $A_{ij} = 1$ on the 100-th sub- and super-diagonals, i.e., for $|i - j| = 100$, and the right hand side $\mathbf{b} = (1, 1, \ldots, 1)^\top$



- Straight CG iteration converges slowly, achieving about 5-digit residual reduction after 40 iterations
- Straight CG is an improvement over a direct method
- Take $M = \mathrm{diag}(A)$, the diagonal matrix with entries $M_{ii} = 0.5 + \sqrt{i}$
- Set $C = \sqrt{M}$ for a new preconditioned CG iteration and with 30 steps it gives convergence to 15-digit residual reduction

Preconditioners

- Reduce condition number
- Sometimes simple, but often they are more complicated
- In various forms with different assumptions
- Effective for eigenvalue problems as well as systems of equations
- See text for more examples

Jacobi preconditioner

- Perhaps the most important preconditioner: $M = \text{diag}(A)$, provided that this matrix is nonsingular
- Also known as diagonal scaling
- More generally, one may take $M = \text{diag}(\mathbf{c})$ for a suitably chosen vector $\mathbf{c} \in \mathbb{C}^m$
- It is a hard mathematical problem to determine a vector $\mathbf{c}$ such that $\kappa(M^{-1}A)$ is exactly minimized
- Fortunately, nothing like the exact minimum is needed in practice

Polynomial preconditioner

Theorem

If A is an $n \times n$ matrix such that $\|A\| < 1$, then $I - A$ is invertible, and

$$(I - A)^{-1} = \sum_{k=0}^{\infty} A^k$$

- It is essential A^{-1} rather than A itself is approximated by the preconditioner
- A polynomial preconditioner is a matrix polynomial $M^{-1} = p(A)$ with the property that $p(A)A$ has better properties for iteration than A itself
- For example, $p(A)$ might be obtained from the first few terms of the Neumann series $A^{-1} = I + (I - A) + (I - A)^2 + \cdots$, or from some other expression, often motivated by approximation theory in the complex plane
- Implemented is based on the same "black box" used for Krylov subspace iteration