Lecture 3 Iterative methods for solving linear system

Weinan E[1,2] and Tiejun Li[2]

[1] Department of Mathematics,
Princeton University,
weinan@princeton.edu

[2] School of Mathematical Sciences,
Peking University,
tieli@pku.edu.cn
No.1 Science Building, 1575

Outline

Iterations

- Iterative methods

  Object: construct sequence $\{\boldsymbol{x}^k\}_{k=1}^{\infty}$, such that $\boldsymbol{x}^k$ converge to a fixed vector $\boldsymbol{x}^*$, and $\boldsymbol{x}^*$ is the solution of the linear system.

- General iteration idea:

  If we want to solve equations

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0},$$

  and the equation $\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x})$ has the same solution as it, then construct

$$\boldsymbol{x}^{k+1} = \boldsymbol{f}(\boldsymbol{x}^k).$$

  If $\boldsymbol{x}^k \longrightarrow \boldsymbol{x}^*$, then $\boldsymbol{x}^* = \boldsymbol{f}(\boldsymbol{x}^*)$, thus the root of $\boldsymbol{g}(\boldsymbol{x})$ is obtained.

Matrix form for Jacobi iterations

- Decompose $\boldsymbol{A} = \boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U}$, where

$$\boldsymbol{L} = \begin{pmatrix} 0 & & & \\ -a_{21} & 0 & & \\ \vdots & \vdots & \ddots & \\ -a_{n1} & -a_{n2} & \cdots & 0 \end{pmatrix}, \quad \boldsymbol{U} = \begin{pmatrix} 0 & -a_{12} & \cdots & -a_{1n} \\ & 0 & \cdots & -a_{2n} \\ & & \ddots & \vdots \\ & & & 0 \end{pmatrix}$$

  and $\boldsymbol{D} = \text{diag}(a_{11}, a_{22}, \ldots, a_{nn})$.

  Transformation: $\boldsymbol{Dx} = \boldsymbol{b} + (\boldsymbol{L} + \boldsymbol{U})\boldsymbol{x}$, thus

$$\boldsymbol{x} = \boldsymbol{D}^{-1}(\boldsymbol{b} + (\boldsymbol{L} + \boldsymbol{U})\boldsymbol{x}) := \boldsymbol{B}\boldsymbol{x} + \boldsymbol{g}.$$

  Here $\boldsymbol{B} = \boldsymbol{D}^{-1}(\boldsymbol{L} + \boldsymbol{U})$ and $\boldsymbol{g} = \boldsymbol{D}^{-1}\boldsymbol{b}$.

- Define the iterations

$$\boldsymbol{x}^{k+1} = \boldsymbol{B}\boldsymbol{x}^{k} + \boldsymbol{g}$$

  This is Jacobi iteration.

Component form of Jacobi iterations

- Jacobi iterations

$$
\begin{aligned}
x_1^{k+1} &= \left(b_1 - \sum_{j\neq 1} a_{1j} x_j^k\right)/a_{11} \\
x_2^{k+1} &= \left(b_2 - \sum_{j\neq 2} a_{2j} x_j^k\right)/a_{22} \\
\cdots \quad &\cdots \quad \cdots \\
x_n^{k+1} &= \left(b_n - \sum_{j\neq n} a_{nj} x_j^k\right)/a_{nn}
\end{aligned}
$$

- Update each component of $\boldsymbol{x}^k$ according to each equation simultaneously and independently by freezing the other variables at former step.
- Parallel in nature!

Jacobi iterations

- Example 1: Solving the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \begin{pmatrix} 1 & 2 & -2 \\ 1 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

- Example 2: Solving the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \begin{pmatrix} 2 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -2 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

Jacobi iterations

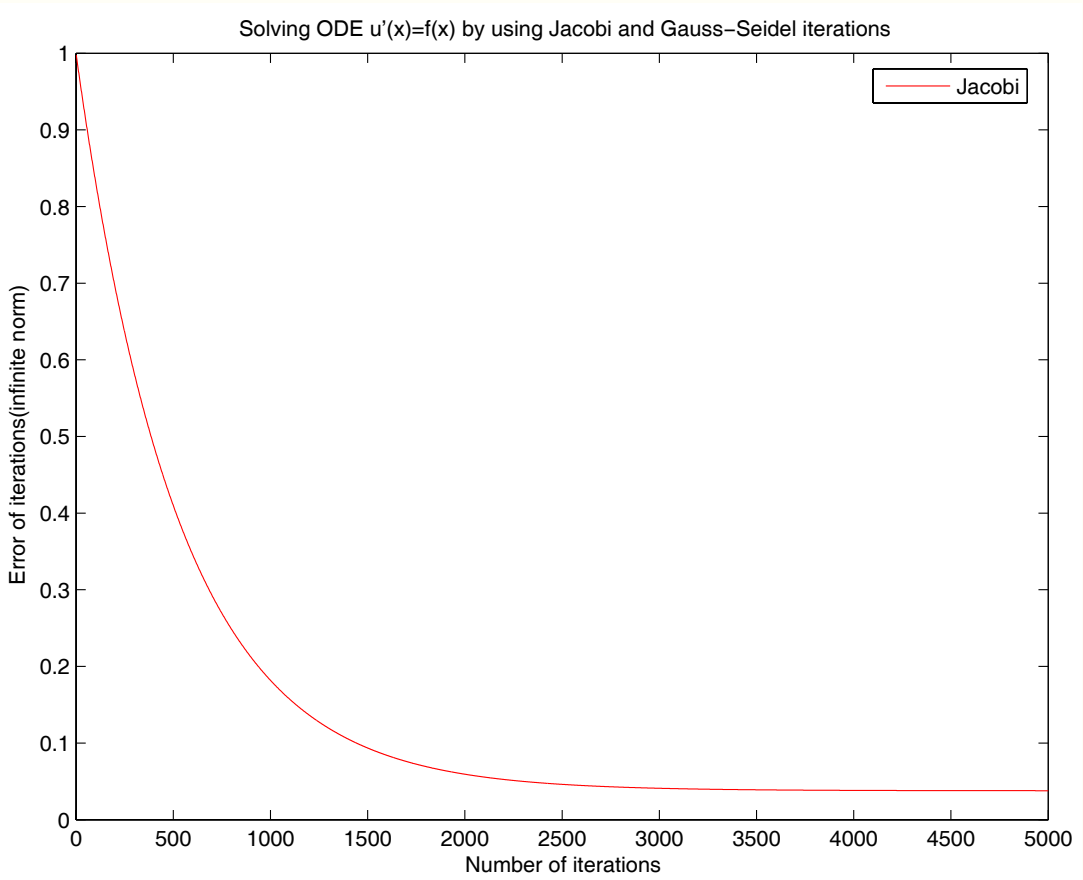
- Example 3: Solving the system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ $(n = 30)$

$$\boldsymbol{A} = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

and $\boldsymbol{b}$ is discretized from $-\pi^2 \sin \pi x,\ x \in [0, 1]$.

Jacobi iterations

- Convergence rate for example 3: (error vs. iteration times)



Gauss-Seidel iterations

- Idea: Update each component of $\boldsymbol{x}^{(k)}$ sequentially with the most updated information and freeze the left components.

$$
\begin{aligned}
x_1^{k+1} &= \left(b_1 - \sum_{j>1} a_{1j} x_j^k\right)/a_{11} \\
x_2^{k+1} &= \left(b_2 - \sum_{j<2} a_{2j} x_j^{k+1} - \sum_{j>2} a_{2j} x_j^k\right)/a_{22} \\
\cdots \quad & \cdots \quad \cdots \\
x_n^{k+1} &= \left(b_n - \sum_{j<n} a_{nj} x_j^{k+1}\right)/a_{nn}
\end{aligned}
$$

- NOT parallel directly.

Matrix form for Gauss-Seidel iteration

- Decompose $\boldsymbol{A} = \boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U}$, and perform transformation $(\boldsymbol{D} - \boldsymbol{L})\boldsymbol{x} = \boldsymbol{b} + \boldsymbol{U}\boldsymbol{x}$, thus

$$\boldsymbol{x} \quad = \quad (\boldsymbol{D} - \boldsymbol{L})^{-1}(\boldsymbol{b} + \boldsymbol{U}\boldsymbol{x}) := \boldsymbol{B}\boldsymbol{x} + \boldsymbol{g}$$

  Here $\boldsymbol{B} = (\boldsymbol{D} - \boldsymbol{L})^{-1}\boldsymbol{U}$ and $\boldsymbol{g} = (\boldsymbol{D} - \boldsymbol{L})^{-1}\boldsymbol{b}$.

- Define the iterations

$$\boldsymbol{x}^{k+1} = \boldsymbol{D}^{-1}\boldsymbol{L}\boldsymbol{x}^{k+1} + \boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{x}^{k} + \boldsymbol{D}^{-1}\boldsymbol{b}$$

  This is Gauss-Seidel iteration.

Gauss-Seidel iterations

- Example 1: Solving the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \begin{pmatrix} 1 & 2 & -2 \\ 1 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

- Example 2: Solving the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \begin{pmatrix} 2 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -2 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

Gauss-Seidel iterations

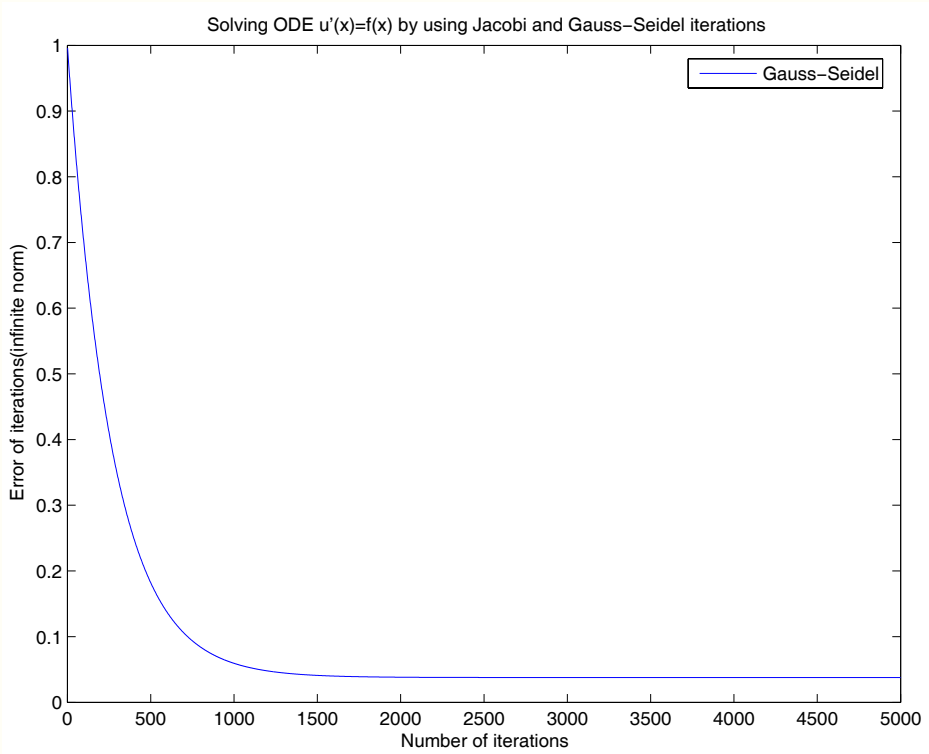
- Example 3: Solving the system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ $(n = 30)$

$$\boldsymbol{A} = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

and $\boldsymbol{b}$ is discretized from $-\pi^2 \sin \pi x,\ x \in [0, 1]$.

Gauss-Seidel iterations

- Convergence rate for example 3: (error vs. iteration times)



Successive relaxation and SOR

- Gauss-Seidel

$$\boldsymbol{x}^{k+1} = \boldsymbol{D}^{-1}\boldsymbol{L}\boldsymbol{x}^{k+1} + \boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{x}^{k} + \boldsymbol{D}^{-1}\boldsymbol{b}.$$

  Define

$$\Delta\boldsymbol{x} = \boldsymbol{x}^{k+1} - \boldsymbol{x}^{k}$$

  then $\boldsymbol{x}^{k+1} = \boldsymbol{x}^{k} + \Delta\boldsymbol{x}$.

- Successive relaxation is to add a weight ω

$$\begin{aligned} \boldsymbol{x}^{k+1} &= \boldsymbol{x}^{k} + \omega\Delta\boldsymbol{x} \\ &= (1-\omega)\boldsymbol{x}^{k} + \omega(\boldsymbol{D}^{-1}\boldsymbol{L}\boldsymbol{x}^{k+1} + \boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{x}^{k} + \boldsymbol{D}^{-1}\boldsymbol{b}). \end{aligned}$$

  If $0 < \omega < 1$, it is called under-relaxation; if $1 < \omega < 2$, it is called over-relaxation;

Convergence theorem

Theorem (Necessary condition)

The necessary condition for the convergence of SR method is

$$0 < \omega < 2$$

Theorem (SPD matrix)

If $\boldsymbol{A}$ is symmetric positive definite, then

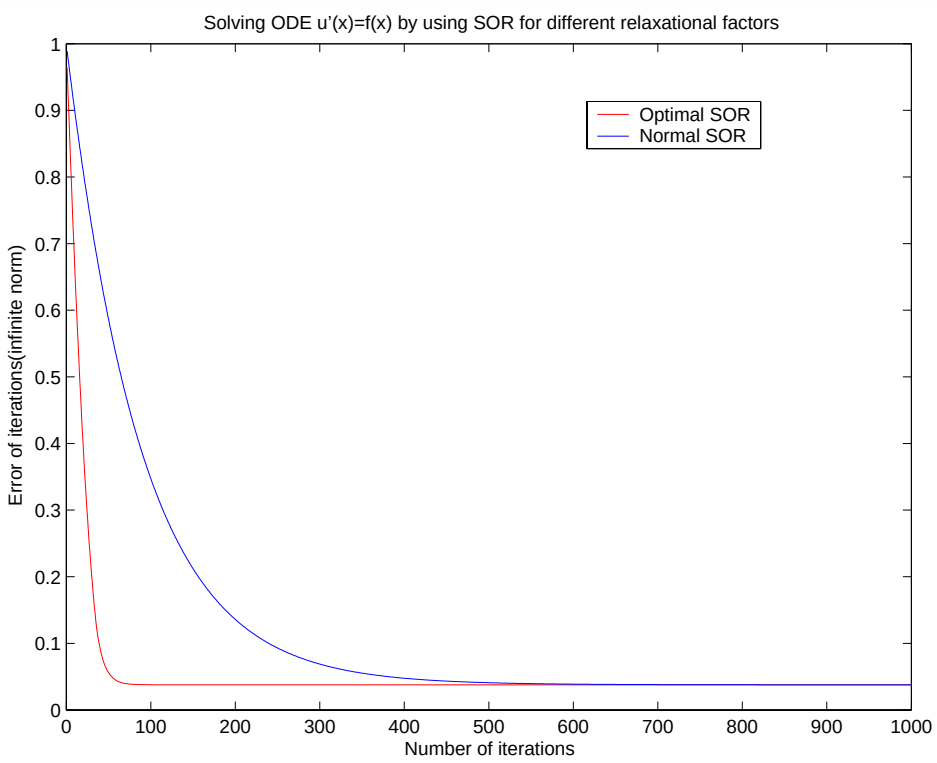
$$0 < \omega < 2$$

is the sufficient condition for convergence.

Examples

SOR method for the ODE problem (n=30).

- $\omega = \frac{2}{1+\sqrt{1-(\rho(J))^2}}$, $\boldsymbol{x}^0 = \mathbf{0}$
- $\omega =$ another one, $\boldsymbol{x}^0 = \mathbf{0}$



Outline

Puzzle?

- Arbitrary decomposition ($\boldsymbol{F}$ is invertible)

$$\boldsymbol{A} = \boldsymbol{F} - \boldsymbol{G}$$

then the linear system is transformed into

$$\boldsymbol{x} = \boldsymbol{F}^{-1}\boldsymbol{G}\boldsymbol{x} + \boldsymbol{F}^{-1}\boldsymbol{b}.$$

and construct the iteration

$$\boldsymbol{x}^{k+1} = \boldsymbol{F}^{-1}\boldsymbol{G}\boldsymbol{x}^{k} + \boldsymbol{F}^{-1}\boldsymbol{b}.$$

How about the result?

- A simpler case (1D case) $ax = b$, $a = f - g$ then construct

$$x^{k+1} = f^{-1}gx^{k} + f^{-1}b.$$

How about the convergence?

Analysis for simple case

- Linear system $ax = b$

  Exact solution $x^* = f^{-1}gx^* + f^{-1}b$

  Iteration $x^{k+1} = f^{-1}gx^k + f^{-1}b$.

- Subtract two expressions and define error $e^k = x^k - x^*$ thus

$$e^{k+1} = f^{-1}ge^k$$

  Then $|e^k| = |f^{-1}g|^k|e^0|$. In order $|e^k| \to 0$, we must have

$$|f^{-1}g| < 1$$

  That's the convergence condition for 1D case.

A fundamental theorem

- Spectral radius

  The spectral radius of a matrix $\boldsymbol{A}$ is defined as

$$\rho(\boldsymbol{A}) = \max_{\lambda} |\lambda(\boldsymbol{A})|$$

Theorem (Spectral radius and iterative convergence)

If $\boldsymbol{A} \in \mathbb{R}^n$, then

$$\lim_{k \to 0} \boldsymbol{A}^k = \mathbf{0} \iff \rho(\boldsymbol{A}) < 1$$

Analysis for general case

- Linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$

  Exact solution $\boldsymbol{x}^* = \boldsymbol{B}\boldsymbol{x}^* + \boldsymbol{g}$

  Iteration $\boldsymbol{x}^{k+1} = \boldsymbol{B}\boldsymbol{x}^k + \boldsymbol{g}$

- Subtract two expressions and define error $\boldsymbol{e}^k = \boldsymbol{x}^k - \boldsymbol{x}^*$ thus

$$\boldsymbol{e}^{k+1} = \boldsymbol{B}\boldsymbol{e}^k$$

  By induction we have

$$\boldsymbol{e}^k = \boldsymbol{B}^k\boldsymbol{e}^0.$$

  Thus we have

$$\boldsymbol{e}^k \to 0 \Longleftrightarrow \boldsymbol{B}^k \to 0 \Longleftrightarrow \rho(\boldsymbol{B}) < 1$$

  That's the convergence criterion for general iterations.

Examples

- Analysis for example 1.

  Spectral radius for Jacobi:

  Spectral radius for Gauss-Seidel:

- Analysis for example 2.

  Spectral radius for Jacobi:

  Spectral radius for Gauss-Seidel:

Rate of convergence

Theorem (Rate of convergence)

If $\|\boldsymbol{B}\| = q < 1$, then the iterated solution has the following convergence rate:

$$\|\boldsymbol{e}^k\| \leq \frac{q^k}{1-q}\|\boldsymbol{x}^1 - \boldsymbol{x}^0\|$$

Loosely speaking, if we ask a tolerance ε_{tol} and $\|\boldsymbol{e}^k\| \leq \varepsilon_{tol}$, we need the number of iterations

$$k \geq \frac{\ln(\varepsilon_{tol}/C)}{\ln q}.$$

where $C = \frac{\|\boldsymbol{x}^1 - \boldsymbol{x}^0\|}{1-q}$.

Remark (Linear convergence)

This type of convergence is called linear convergence, which means

$$\lim_{k\to\infty} \frac{\|\boldsymbol{e}^{k+1}\|}{\|\boldsymbol{e}^k\|} = C \neq 0.$$

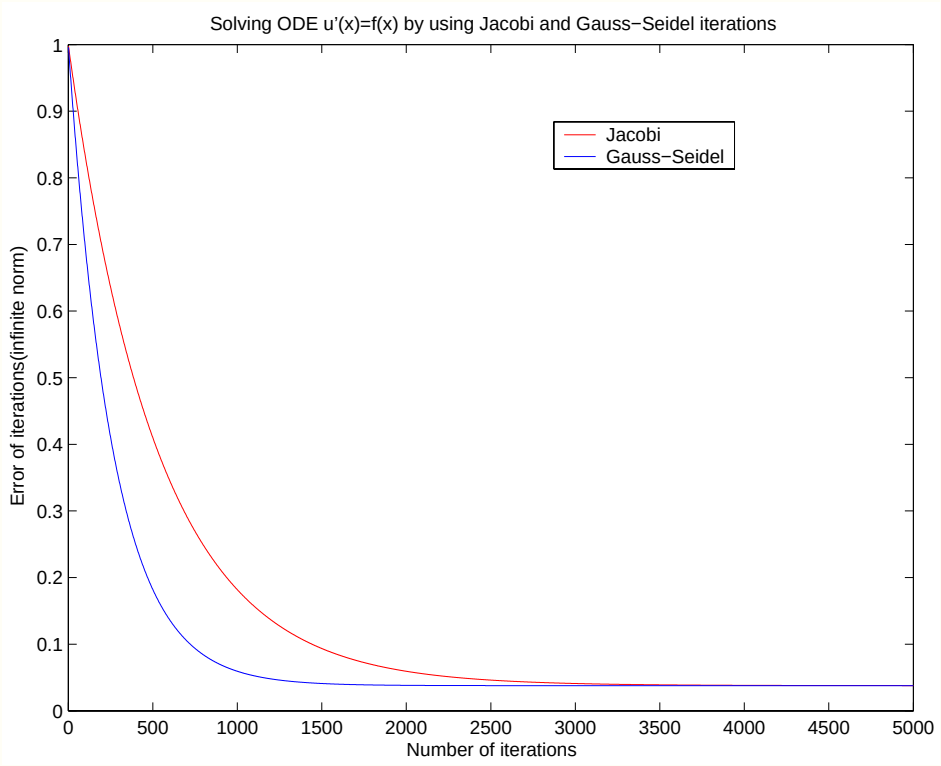
Examples

- Convergence rate for example 3

  Spectral radius for Jacobi:

  Spectral radius for Gauss-Seidel:

- Comparison of numerical convergence rate:



Diagonally dominant matrix and its properties

Definition (Diagonally dominant matrix (DDM))

Suppose $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, *if*

$$|a_{ii}| > \sum_{j=1, j \neq i}^{n} |a_{ij}|, \quad i = 1, \ldots, n$$

$\boldsymbol{A}$ *is called a DDM.*

- DDM Example

$$\boldsymbol{A} = \begin{pmatrix} 3 & -1 & 1 \\ 1 & 3 & 1 \\ 0 & 1 & -4 \end{pmatrix}$$

- Example 3 does NOT belong to this class.

Diagonally dominant matrix and its properties

Theorem (Convergence for DDM)

If $\boldsymbol{A}$ is a DDM, $\boldsymbol{A}$ is nonsingular and the Jacobi and Gauss-Seidel method for $\boldsymbol{Ax} = \boldsymbol{b}$ is convergent.

Outline

Symmetric positive definite (SPD) matrix and Quadratic function

- Linear system

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

  where $\boldsymbol{A}$ is a SPD matrix.

- Quadratic function

$$\varphi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{A}\boldsymbol{x} - b^T \boldsymbol{x}$$

- Connection

$$\text{Solving } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \Longleftrightarrow \min \varphi(\boldsymbol{x})$$

Symmetric positive definite (SPD) matrix and Quadratic function

- Basic observation

$$\frac{\partial \varphi}{\partial x_i} = \sum_{j=1}^{n} a_{ij} x_j - x_i b_i$$

Define gradient vector

$$\text{grad}\varphi = \left(\frac{\partial \varphi}{\partial x_i} \right)_{i=1}^{n} = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$$

The minimizer $\boldsymbol{x}^*$ of φ satisfies

$$\text{grad}\varphi = 0, \text{ i.e. } \boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}$$

- The necessity is skipped.

Symmetric positive definite (SPD) matrix and Quadratic function

The connection between linear system and quadratic function minimization tells us if we have an algorithm to deal with quadratic function minimization we have an algorithm for solving the linear system.

Steepest decent for quadratic minimization

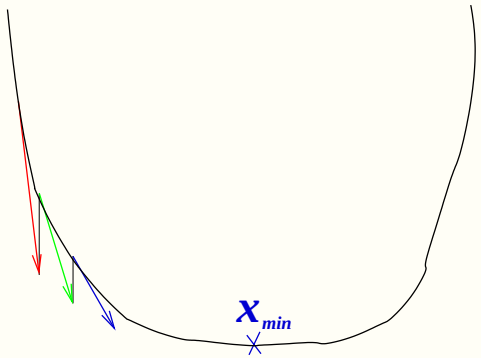
- Basic idea: Find a series of decent directions $\boldsymbol{p}^k$ and corresponding stepsize α_k such that the iterations

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{p}^k$$

and

$$\varphi(\boldsymbol{x}^{k+1}) \leq \varphi(\boldsymbol{x}^k).$$

- Schematics for Steepest decent



- How to find $\boldsymbol{p}^k$ and α_k?

Steepest decent method (SDM) for quadratic minimization

- The negative gradient direction $-\text{grad}\varphi$ is the "steepest" decent direction, so choose

$$\boldsymbol{p}^k = \boldsymbol{r}^k := -\text{grad}\varphi(\boldsymbol{x}^k)$$

and choose α_k such that

$$\min_{\alpha} \varphi(\boldsymbol{x}^k + \alpha \boldsymbol{p}^k)$$

- Define $f(\alpha) = \varphi(\boldsymbol{x}^k + \alpha \boldsymbol{p}^k)$ then

$$\frac{\partial f(\alpha)}{\partial \alpha} = 0.$$

We obtain

$$\alpha_k = \frac{(\boldsymbol{r}^k)^T \boldsymbol{p}^k}{(\boldsymbol{p}^k)^T \boldsymbol{A} \boldsymbol{p}^k}$$

where $\boldsymbol{r}^k = -\text{grad}\varphi(\boldsymbol{x}^k) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k$.

- This is the so called steepest decent method.

Example

- Example 1:

$$\boldsymbol{A} = \begin{pmatrix} 4 & 1 & 1 & 0 \\ 1 & 4 & 1 & 1 \\ 1 & 1 & 4 & 1 \\ 0 & 1 & 1 & 4 \end{pmatrix} \quad \boldsymbol{b} = \begin{pmatrix} 6 \\ 7 \\ 7 \\ 6 \end{pmatrix}$$

- The ODE example (n=30):

  Convergence rate.

Convergence of steepest decent method

Theorem (Convergence of steepest decent method)

Suppose the eigenvalues of $\boldsymbol{A}$ *are* $0 < \lambda_1 \leq \cdots \leq \lambda_n$*, then the iterating sequence* $\{\boldsymbol{x}^k\}_{k=1}^{\infty}$ *by steepest decent method has the following convergence property*

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{A}} \leq \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_{\boldsymbol{A}}$$

where $\boldsymbol{x}^* = \boldsymbol{A}^{-1}\boldsymbol{b}$, $\|\boldsymbol{x}\|_{\boldsymbol{A}} := \sqrt{\boldsymbol{x}^T \boldsymbol{A}\boldsymbol{x}}$.

Remarks on steepest decent method

Remark

1. *Consider* $\boldsymbol{p}^k = -grad\varphi(\boldsymbol{x}^k), \boldsymbol{p}^{k+1} = -grad\varphi(\boldsymbol{x}^{k+1}), \boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{p}^k$, *we have*

$$\frac{d}{d\alpha}\varphi(\boldsymbol{x}^k + \alpha \boldsymbol{p}^k)\Big|_{\alpha=\alpha_k} = 0$$

i.e.

$$(\boldsymbol{p}^{k+1})^T \boldsymbol{p}^k = 0$$

the decent directions are orthogonal for two neighboring steps. This means steepest decent is locally steepest decent, not globally steepest decent! This effect becomes quite severe when $\boldsymbol{x}^k$ *approaches* $\boldsymbol{x}^*$.

2. *If* $\lambda_n \gg \lambda_1$, *then* $\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} \approx 1$, *the convergence rate will be very slow!*
3. *Though the steepest decent method is easily implemented, it is very rarely used for the drawback stated above.*
4. *It can be slightly modified to obtain the famous conjugate gradient method.*

Conjugate gradient method (CGM)

- Only modify SDM a little bit to obtain CGM.
- Suppose we have $\boldsymbol{x}^k, \boldsymbol{p}^{k-1}, \boldsymbol{r}^k$. The strategy of CGM is not to choose $\boldsymbol{p}^k = \boldsymbol{r}^k$, but choose $\boldsymbol{p}^k$ in 2D plane

$$\pi_2 := \{\xi \boldsymbol{r}^k + \eta \boldsymbol{p}^{k-1}, \xi, \eta \in \mathbb{R}\}$$

Define $\psi(\xi, \eta) = \varphi(\boldsymbol{x}_k + \xi \boldsymbol{r}^k + \eta \boldsymbol{p}^{k-1})$, then take

$$\frac{\partial \psi}{\partial \xi} = 0, \quad \frac{\partial \psi}{\partial \eta} = 0$$

then

$$\begin{cases} \xi_0 (\boldsymbol{r}^k)^T \boldsymbol{A} \boldsymbol{r}^k + \eta_0 (\boldsymbol{r}^k)^T \boldsymbol{A} \boldsymbol{p}^{k-1} &= (\boldsymbol{r}^k)^T \boldsymbol{r}^k \\ \xi_0 (\boldsymbol{r}^k)^T \boldsymbol{A} \boldsymbol{p}^{k-1} + \eta_0 (\boldsymbol{p}^{k-1})^T \boldsymbol{A} \boldsymbol{p}^{k-1} &= \boldsymbol{0} \end{cases}$$

i.e. we can take

$$\boldsymbol{p}^k = \boldsymbol{r}_k + \frac{\eta_0}{\xi_0} \boldsymbol{p}^{k-1}$$

- Same as SDM to obtain α_k and $\boldsymbol{x}^{k+1}$.

Conjugate gradient method (CGM)

Final formulation of CGM:

1. Initial step: $\boldsymbol{x}^0, \boldsymbol{p}^0 = \boldsymbol{r}^0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^0$
2. Suppose we have $\boldsymbol{x}^k, \boldsymbol{r}^k, \boldsymbol{p}^k$, the CGM step
   - 2.1 Search the optimal α_k along $\boldsymbol{p}^k$;

$$\alpha_k = \frac{(\boldsymbol{r}^k)^T \boldsymbol{p}^k}{(\boldsymbol{p}^k)^T \boldsymbol{A}\boldsymbol{p}^k}$$

   - 2.2 Update $\boldsymbol{x}^k$ and gradient direction $\boldsymbol{r}^k$;

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{p}^k, \quad \boldsymbol{r}^{k+1} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{k+1}$$

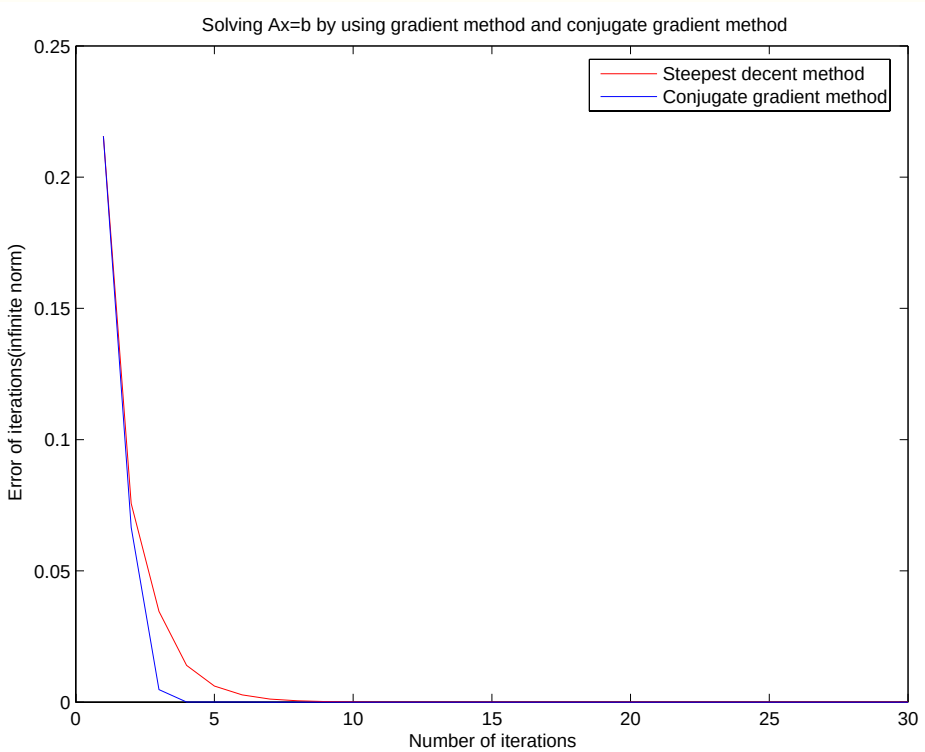
   - 2.3 According to the calculation before to form new search direction $\boldsymbol{p}^{k+1}$

$$\beta_k = -\frac{(\boldsymbol{r}^{k+1})^T \boldsymbol{A}\boldsymbol{p}^k}{(\boldsymbol{p}^k)^T \boldsymbol{A}\boldsymbol{p}^k}, \quad \boldsymbol{p}^{k+1} = \boldsymbol{r}^{k+1} + \beta_k \boldsymbol{p}^k$$

Example

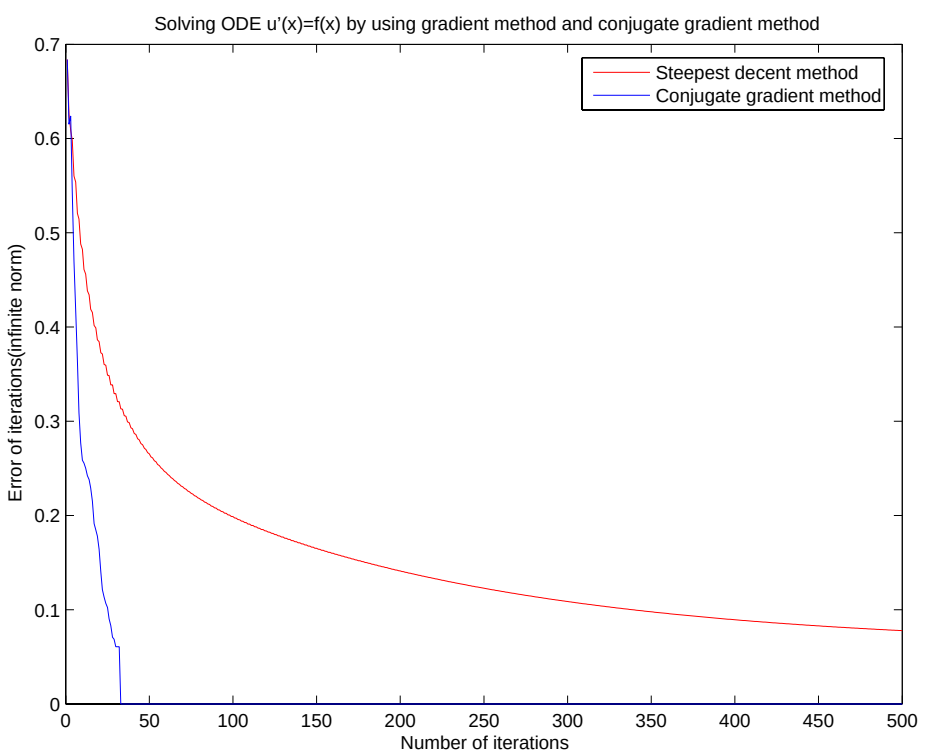
- Example 1:

$$\boldsymbol{A} = \begin{pmatrix} 4 & 1 & 1 & 0 \\ 1 & 4 & 1 & 1 \\ 1 & 1 & 4 & 1 \\ 0 & 1 & 1 & 4 \end{pmatrix} \quad \boldsymbol{b} = \begin{pmatrix} 6 \\ 7 \\ 7 \\ 6 \end{pmatrix}$$



Example

- The ODE example (n=30):



Properties of CGM

Theorem

The vectors generated by CGM have the properties

1. $(\boldsymbol{r}^i)^T \boldsymbol{r}^j = 0, \quad i \neq j,\ 0 \leq i < j \leq k;$
2. $(\boldsymbol{p}^i)^T \boldsymbol{A}\boldsymbol{p}^j = 0, \quad i \neq j,\ 0 \leq i < j \leq k;$

- The property $(\boldsymbol{p}^i)^T \boldsymbol{A}\boldsymbol{p}^j = 0$ makes $\{\boldsymbol{p}^j\}_{j=1,2,\ldots}$ are called conjugate directions of $\boldsymbol{A}$. From property 1, we have

  Theoretically, CGM will find the minimum (or solution) within n steps.

  This is not the case in real computations. CGM is applied as an iteration.

- Virtue of CGM: Suitable for sparse matrix $\boldsymbol{A}$, without adjustable parameters as SOR.

Convergence of CGM

Theorem

The sequence $\{\boldsymbol{x}^k\}$ *has the error estimate*

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{A}} \leq 2\Big(\frac{\sqrt{\kappa_2}-1}{\sqrt{\kappa_2}+1}\Big)\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_{\boldsymbol{A}}$$

where $\kappa_2 := Cond_2(\boldsymbol{A}) = \|\boldsymbol{A}\|_2\|\boldsymbol{A}^{-1}\|_2$.

This theorem tells us if the matrix in good condition ($Cond_2(\boldsymbol{A}) \approx 1$), the convergence is very fast. The real computations are even faster than the estimate.

Preconditioning conjugate gradient method (PCG)

- If the conditioning of $\boldsymbol{A}$ is bad, we need the so-called preconditioning technique.
- Formally transform $\boldsymbol{Ax} = \boldsymbol{b}$ into

$$\hat{\boldsymbol{A}}\hat{\boldsymbol{x}} = \hat{\boldsymbol{b}}$$

  where $\hat{\boldsymbol{A}} = \boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P}^{-1}, \hat{\boldsymbol{x}} = \boldsymbol{P}\boldsymbol{x}, \hat{\boldsymbol{b}} = \boldsymbol{P}^{-1}\boldsymbol{b}$, $\boldsymbol{P}$ is a SPD matrix.
- Apply CGM to the modified system to obtain $\hat{\boldsymbol{x}}^k, \hat{\boldsymbol{r}}^k, \hat{\boldsymbol{p}}^k$.
- The matrix $\boldsymbol{M} := \boldsymbol{P}^2$ is called preconditioner. Usually $\boldsymbol{M}$ is chosen a sparse SPD matrix, and the eigenvalues of $\boldsymbol{M}^{-1}\boldsymbol{A}$ concentrate on some value, and the equation $\boldsymbol{My} = \boldsymbol{r}$ is easy to be solved.

Extension to general linear system

- Technique 1: Regularization. Transform the linear system

$$\boldsymbol{A}^T \boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}\boldsymbol{b}$$

  If the conditioning of $\boldsymbol{A}$ is bad, the convergence is very slow.

- Generalized Minimum RESidual method (GMRES):
  Compute the minimal residual problem

$$\min\{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2, \boldsymbol{x} \in \boldsymbol{x}_0 + \text{ some subspace}\}$$

  This formulation is well-posed for non-symmetric matrix.

- References:
  Y. Saad, Iterative methods for sparse linear systems, PWS Publishing Company, 1996.

Homework assignment 3

1. Using Gauss-Seidel and conjugate gradient method to solve the second order ODEs(n=30, 50, 100). Plot the figure for the convergence rate.