Solution of Algebraic Equations

- Systems of linear equations
- Direct method
  - Gauss elimination
  - Doolittle and Crout method
  - Modified Cholesky method
- Iterative method
  - Jacobi method and Gauss-Seidel method
  - Steepest descent method and Conjugate Gradient method

Solution Method

$$\boldsymbol{Ax} = \boldsymbol{b} \xrightarrow{\text{regular } \boldsymbol{A}} = \boldsymbol{A}^{-1}\boldsymbol{b} \quad \left(\text{Cramer's rule? } (n+1)(n-1)n! + n\right)$$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \qquad\qquad \vdots \qquad\qquad\qquad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

- Direct method
  - Unconditionally solve
- Iterative method
  - Appropriate for large-scale problem
  - Initial value assumption, Ill-conditioned matrix

Direct Method

- Gauss elimination
- Doolittle-Crout method
- Cholesky method
- Modified Cholesky method

$$\boldsymbol{A} = \begin{cases} \boldsymbol{LU} \\ \boldsymbol{LL}^T \\ \boldsymbol{LDL}^T \end{cases} \rightarrow \boldsymbol{Ax} = \boldsymbol{b} \rightarrow \begin{cases} \boldsymbol{L}\underbrace{\boldsymbol{Ux}}_{\boldsymbol{y}} = \boldsymbol{b} \\ \boldsymbol{L}\underbrace{\boldsymbol{L}^T\boldsymbol{x}}_{\boldsymbol{y}} = \boldsymbol{b} \\ \boldsymbol{L}\underbrace{\boldsymbol{DL}^T\boldsymbol{x}}_{\boldsymbol{y}} = \boldsymbol{b} \end{cases} \rightarrow \boldsymbol{Ly} = \boldsymbol{b} \rightarrow \begin{cases} \boldsymbol{Ux} = \boldsymbol{y} \\ \boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{y} \\ \boldsymbol{DL}^T\boldsymbol{x} = \boldsymbol{y} \end{cases}$$

Gauss Elimination: Forward Reduction

$$\begin{cases} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + \cdots + a_{1n}^{(0)}x_n = b_1^{(0)} \cdots\cdots(1) \rightarrow x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ a_{21}^{(0)}x_1 + a_{22}^{(0)}x_2 + \cdots + a_{2n}^{(0)}x_n = b_2^{(0)} \cdots\cdots(2) \\ \qquad\qquad \vdots \qquad\qquad\qquad \vdots \\ a_{n1}^{(0)}x_1 + a_{n2}^{(0)}x_2 + \cdots + a_{nn}^{(0)}x_n = b_n^{(0)} \cdots\cdots(n) \end{cases}$$

$$\begin{cases} x_1 + a_{12}^{(1)}x_2 + \cdots + a_{1n}^{(1)}x_n = b_1^{(1)} \cdots\cdots(1') \quad \text{where} \begin{cases} a_{1j}^{(1)} = \dfrac{a_{1j}^{(0)}}{a_{11}^{(0)}} \quad (j = 1, 2, \cdots, n) \\ b_1^{(1)} = \dfrac{b_1^{(0)}}{a_{11}^{(0)}} \end{cases} \\ \left.\begin{matrix} a_{22}^{(1)}x_2 + \cdots + a_{2n}^{(1)}x_n = b_2^{(1)} \cdots\cdots(2') \\ \vdots \qquad\qquad \vdots \\ a_{n2}^{(1)}x_2 + \cdots + a_{nn}^{(1)}x_n = b_n^{(1)} \cdots\cdots(n') \end{matrix}\right\} \quad \text{where} \begin{cases} a_{ij}^{(1)} = a_{ij}^{(0)} - a_{i1}^{(0)}a_{1j}^{(1)} \quad (i, j = 2, 3, \cdots, n) \\ b_i^{(1)} = b_i^{(0)} - a_{i1}^{(0)}b_1^{(1)} \quad (i = 2, 3, \cdots, n) \end{cases} \end{cases}$$

Gauss Elimination: Backward Substitution

$$\left[k\text{-th row: }(\text{diagonal})\text{ coefficient of }x_k \rightarrow 1\right]$$

$$\begin{cases} a_{kj}^{(k)} = a_{kj}^{(k-1)} \Big/ a_{kk}^{(k-1)} & (j = k, k+1, \cdots, n) \\ b_k^{(k)} = b_i^{(k-1)} \Big/ a_{kk}^{(k-1)} & (i = 2, 3, \cdots, n) \end{cases}$$

$$\left[(k+1)\text{-th row: coefficient of }x_k \rightarrow 0\right]$$

$$\begin{cases} a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} a_{kj}^{(k)} & (j = k+1, \cdots, n) \\ b_i^{(k)} = b_i^{(k-1)} - a_{ik}^{(k-1)} b_k^{(k)} \end{cases}$$

$$\begin{cases} \left.\begin{array}{r} x_1 + a_{12}^{(1)} x_2 + a_{13}^{(1)} x_3 + \cdots + a_{1,n-1}^{(1)} x_{n-1} + a_{1n}^{(1)} x_n = b_1^{(1)} \\ x_2 + a_{23}^{(2)} x_3 + \cdots + a_{2,n-1}^{(2)} x_{n-1} + a_{2n}^{(2)} x_n = b_2^{(2)} \\ x_3 + \cdots + a_{3,n-1}^{(3)} x_{n-1} + a_{3n}^{(3)} x_n = b_3^{(3)} \end{array}\right\} \rightarrow x_k = b_k^{(k)} - \sum_{j=k+1}^{n} a_{kj}^{(k)} x_j \\ \qquad\qquad \vdots \qquad \vdots \\ x_{n-1} + a_{n-1,n}^{(n-1)} x_n = b_3^{(3)} \rightarrow x_{n-1} = b_{n-1}^{(n-1)} - a_{n-1,n}^{(n-1)} x_n \\ a_{nn}^{(n-1)} x_n = b_n^{(n-1)} \rightarrow x_n = \dfrac{b_n^{(n-1)}}{a_{nn}^{(n-1)}} \end{cases}$$

Example 1

$$\begin{cases} 2x_1 + 3x_2 + 4x_3 = 17 \\ 3x_1 + 5x_2 - 2x_3 = 10 \\ 4x_1 - 2x_2 + x_3 = 12 \end{cases} \rightarrow \begin{cases} x_1 = 3 \\ x_2 = 1 \\ x_3 = 2 \end{cases}$$

Doolittle-Crout Method

$$\boldsymbol{A} = \boldsymbol{LU} \quad \text{where} \quad \boldsymbol{L} = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & 0 & \\ l_{31} & l_{32} & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & 1 \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{12} & u_{13} & \cdots & u_{1n} \\ & & u_{13} & \cdots & u_{1n} \\ & 0 & & \ddots & \vdots \\ & & & & u_{nn} \end{bmatrix}$$

$$\boldsymbol{Ax} = \boldsymbol{b} \rightarrow \boldsymbol{LUx} = \boldsymbol{b} \xrightarrow{\boldsymbol{Ux}=\boldsymbol{y}} \boldsymbol{Ly} = \boldsymbol{b}$$

$$\boldsymbol{y}(\text{forward}) \rightarrow \boldsymbol{x}(\text{backward})$$

Cholesky Method

$$A = LL^T \Leftrightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & & & 0 \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ & l_{22} & \cdots & l_{n2} \\ & & \ddots & \vdots \\ 0 & & & l_{nn} \end{bmatrix}$$

$$\boldsymbol{Ax} = \boldsymbol{b} \rightarrow \boldsymbol{LL}^T \boldsymbol{x} = \boldsymbol{b} \xrightarrow{\boldsymbol{L}^T \boldsymbol{x} = \boldsymbol{y}} \boldsymbol{Ly} = \boldsymbol{b}$$

$$\boldsymbol{y}(\text{forward}) \rightarrow \boldsymbol{x}(\text{backward})$$

$$\begin{cases} l_{11} = \sqrt{a_{11}} \\ l_{i1} = a_{i1} / l_{11} \quad (i = 2, \cdots, n) \\ l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2} \quad (i = 2, \cdots, n) \\ l_{ij} = \left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk} \right) \Big/ l_{ii} \quad (i = 2, \cdots, n;\ i < j) \end{cases} \Rightarrow \left| \begin{array}{l} \begin{cases} y_1 = b_1 / l_{11} \\ y_i = \left(b_i - \sum_{k=1}^{i-1} l_{ik} y_k \right) \Big/ l_{ii} \quad (i = 2, \cdots, n) \end{cases} \\ \begin{cases} x_n = y_n / l_{nn} \\ x_i = \left(y_i - \sum_{k=i+1}^{n} l_{ki} x_k \right) \Big/ l_{ii} \quad (i = n-1, \cdots, 1) \end{cases} \end{array} \right.$$

Modified Cholesky Method

$$\boldsymbol{A} = \boldsymbol{LDL}^T \Leftrightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & & 0 \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} d_{11} & & & 0 \\ & d_{22} & & \\ & & \ddots & \\ 0 & & & d_{nn} \end{bmatrix} \begin{bmatrix} 1 & l_{21} & \cdots & l_{n1} \\ & 1 & \cdots & l_{n2} \\ & & \ddots & \vdots \\ 0 & & & 1 \end{bmatrix}$$

$$\boldsymbol{Ax} = \boldsymbol{b} \rightarrow \boldsymbol{LDL}^T \boldsymbol{x} = \boldsymbol{b} \xrightarrow{\boldsymbol{DL}^T \boldsymbol{x} = \boldsymbol{y}} \boldsymbol{Ly} = \boldsymbol{b}$$

$$\boldsymbol{y}\left(\text{forward}\right) \rightarrow \boldsymbol{x}\left(\text{backward}\right)$$

$$\left.\begin{aligned} a_{ij} &= \sum_{k=1}^{i} l_{ik} d_{kk} l_{jk} \quad \left(i = 1, \cdots, n,\ j = 1, \cdots, i-1\right) \\ a_{ii} &= \sum_{k=1}^{i} l_{ik} d_{kk} l_{ik} \quad \left(i = 1, \cdots, n\right) \end{aligned}\right\} \xrightarrow{l_{ii} = 1} \begin{cases} l_{ij} d_{jj} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} d_{kk} l_{jk} \\ d_{ii} = a_{ii} - \sum_{k=1}^{i-1} l_{ik}{}^2 d_{kk} \end{cases}$$

$$\begin{cases} y_1 = b_1,\ y_i = b_i - \sum_{k=1}^{j-1} l_{ik} y_k \quad \left(i = 2, \cdots, n\right) \\ x_n = \dfrac{y_n}{d_{nn}},\ x_i = \dfrac{y_i}{d_{ii}} - \sum_{k=i+1}^{n} l_{ki} x_k \quad \left(i = n-1, \cdots, 1\right) \end{cases}$$

Example 2

$$\begin{cases} 2x_1 + x_2 + x_3 = 8 \\ x_1 + 3x_2 + 2x_3 = 11 \\ x_1 + 2x_2 + 4x_3 = 16 \end{cases} \rightarrow \begin{cases} x_1 = 2 \\ x_2 = 1 \\ x_3 = 3 \end{cases}$$

Iterative Method

- Jacobi method
- Gauss-Seidel method
- SOR (Successive Over-Reduction) method
- Steepest descent method
- CG (Conjugate Gradient) method
- PCG (Preconditioned Conjugate Gradient) method
- ICCG (Incomplete Cholesky Conjugate Gradient) method

Jacobi → Gauss-Seidel → SOR

$$\begin{cases} 4x_1 + x_2 = 6 \\ x_1 + 5x_2 = 11 \end{cases} \Rightarrow \begin{cases} x_1 = \frac{1}{4}(6 - x_2) \\ x_2 = \frac{1}{5}(11 - x_1) \end{cases}$$

$$[\text{Jacobi method}] \; x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_i^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_i^{(k)} \right) \Big/ a_{ii}$$

$$\rightarrow \text{convergence condition: } \max \left\{ \sum_{j=1}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| \right\} < 1 \quad (i \neq j)$$

$$[\text{Gauss-Seidel method}] \; x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_i^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_i^{(k)} \right) \Big/ a_{ii}$$

$$[\text{SOR(Successive Over-Relaxation method}] \; x_i^{(k+1)} = \omega \tilde{x}_i^{(k+1)} + (1 - \omega) x_i^{(k)}, \; 0 < \omega < 2$$

$$(\omega = 1 : \text{Gauss-Seidel})$$

Example 3

$$\begin{cases} 4x_1 + \ \ x_2 = 6 \\ x_1 + 5x_2 = 11 \end{cases} \Rightarrow \begin{cases} x_1 = \dfrac{1}{4}(6 - x_2) \\ x_2 = \dfrac{1}{5}(11 - x_1) \end{cases}$$

- Jacobi method

iter	0	1	2	3	4	5
x_1	0	1.5	0.95	1.025	0.9975	1.0010
x_2	0	2.2	1.90	2.010	1.9950	2.0005

- Gauss-Seidel method

iter	0	1	2	3	4	5
x_1	0	1.5	1.025	1.001	1.0001	1.000025
x_2	-	1.9	1.995	1.998	1.9999	1.999995

Steepest Decent → Conjugate Gradient

$$\boldsymbol{Ax} = \boldsymbol{b} \Leftrightarrow \min_{\boldsymbol{x}} f\left(\boldsymbol{x}\right) = \frac{1}{2}\left(\boldsymbol{x}, \boldsymbol{Ax}\right) - \left(\boldsymbol{x}, \boldsymbol{b}\right) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{Ax} - \boldsymbol{x}^T \boldsymbol{b}$$

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha^{(k)} \boldsymbol{p}^{(k)}$$

$$\boldsymbol{p}^{(k)} = \begin{cases} -\dfrac{\partial f\left(\boldsymbol{x}^{(k)}\right)}{\partial \boldsymbol{x}^{(k)}} = \boldsymbol{b} - \boldsymbol{Ax}^{(k)} = \boldsymbol{r}^{(k)} & \left[\text{steepest descent method}\right] \\ \boldsymbol{r}^{(k)} + \beta^{(k-1)} \boldsymbol{p}^{(k-1)} & \left[\text{conjugate gradient method}\right] \xrightarrow{\left(\boldsymbol{p}^{(k)}, \boldsymbol{Ap}^{(k-1)}\right) = 0} \beta^{(k)} = -\dfrac{\left(\boldsymbol{r}^{(k+1)}, \boldsymbol{Ap}^{(k)}\right)}{\left(\boldsymbol{p}^{(k)}, \boldsymbol{Ap}^{(k)}\right)} \end{cases}$$

$$\frac{\partial f\left(\boldsymbol{x}^{(k)} + \alpha^{(k)} \boldsymbol{p}^{(k)}\right)}{\partial \alpha^{(k)}} = 0 \rightarrow \alpha^{(k)} = \frac{\left(\boldsymbol{p}^{(k)}, \boldsymbol{r}^{(k)}\right)}{\left(\boldsymbol{p}^{(k)}, \boldsymbol{Ap}^{(k)}\right)}$$

- Fast convergence, less memory
  - PCC(Preconditioned Conjugate Gradient) method
  - ICCG(Incomplete Cholesky Conjugate Gradient) method
- Advantage for parallel computing

Example 4

$$\begin{cases} 4x_1 + \ \ x_2 = 6 \\ \ \ x_1 + 5x_2 = 11 \end{cases} \qquad \boldsymbol{x}^{(0)} = \{x_1 \quad x_2\}^T = \{0 \quad 0\}^T$$