计算方法

第5章 线性方程组的解法

胡 敏
合肥工业大学 计算机与信息学院
jsjxhumin@hfut.edu.cn

第 5 章　线性方程组的解法

5.1 迭代公式的建立

5.2 向量和矩阵的范数

5.3 迭代过程的收敛性

第五章　线性方程组的迭代法

1．教学内容：

首先通过例子介绍解线性方程组的迭代法的基本思想；然后介绍雅可比迭代公式及其程序设计；介绍高斯−塞德尔迭代公式；超松驰迭代法及其程序设计；以及迭代公式的矩阵表示。

2．重点难点：

雅可比迭代法、高斯——塞德尔迭代法、超松驰迭代法

3．教学目标：

掌握三种迭代公式，能利用这三种迭代公式进行线性方程组的迭代求解，并编制相应的应用程序。

引 言

前几章中不管插值公式与求积公式的建立，还是常微分方程差分格式的构造，其基本思想都是将其转化为代数问题来处理，最后归结为解线性方程组。工程技术的科学计算中，线性方程组也会经常遇到。因此，线性方程组的解法在数值分析中占有极其重要的地位。

线性方程组的解法大致分为直接法和迭代法两大类。

直接方法的特点是：如果不考虑计算过程中的舍入误差，运用此类方法经过有限次算术运算就能求出线性方程组的精确解。

迭代法：设计迭代公式，反复进行，无限次求得精确解

迭代法优点：算法简单，因而编制程序比较容易。

迭代法缺点,它要求方程组的系数矩阵具有某种特殊性质，以保证迭代过程的收敛性。发散的迭代过程是没有实用价值的。

5.1 迭代公式的建立

1、雅可比迭代公式

解线性方程组迭代法的基本思想是将联立方程组的求解归结为重复计算一组彼此独立的线性表达式，这就使问题得到了简化。

求解一般方程 $f(x)=0$ 的迭代法的基本想法是：将 $f(x)=0$ 改写为等价的 $x=\varphi(x)$，从而得到迭代格式：$x_{k+1}=\varphi(x_k)$，当 $x_{n+1}=x_n$ 时 $|x_{k+1}-x_k|$ 或很小时，就得到原方程的解。

类似地，可以将这种思想用到方程组的求解上，从而得到线性方程组的迭代解法。

例1：

$$\begin{bmatrix} 10 & -1 & -2 \\ -1 & 10 & -2 \\ -1 & -1 & 5 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 7.2 \\ 8.3 \\ 4.2 \end{bmatrix}$$ 精确解为 $$\begin{bmatrix} 1.1 \\ 1.2 \\ 1.3 \end{bmatrix}$$

解：

把 x_1，x_2，x_3，从三个方程中分离出来得：

$$\begin{cases} x_1 = 0.1x_2 + 0.2x_3 + 0.72 \\ x_2 = 0.1x_1 + 0.2x_3 + 0.83 \\ x_3 = 0.2x_1 + 0.2x_2 + 0.84 \end{cases}$$

记 $$B=\begin{bmatrix} 0 & 0.1 & 0.2 \\ 0.1 & 0 & 0.2 \\ 0.2 & 0.2 & 0 \end{bmatrix}$$ $$f=\begin{bmatrix} 0.72 \\ 0.83 \\ 0.84 \end{bmatrix}$$

则：$x = Bx + f$

得迭代格式：$x^{(n+1)} = Bx^{(n)} + f$

$$\begin{cases} x_1^{k+1} = 0.1x_2^k + 0.2x_3^k + 0.72 \\ x_2^{k+1} = 0.1x_1^k + 0.2x_3^k + 0.83 \\ x_3^{k+1} = 0.2x_1^k + 0.2x_2^k + 0.84 \end{cases}$$

取初值 $x^{(0)} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$

$$x^{(1)} = \begin{bmatrix} 0.72 \\ 0.83 \\ 0.84 \end{bmatrix}, \quad x^{(2)} = \begin{bmatrix} 0.971 \\ 1.070 \\ 1.150 \end{bmatrix}, \quad \cdots\cdots, \quad x^{(9)} = \begin{bmatrix} 1.09994 \\ 1.19994 \\ 1.2992 \end{bmatrix}$$

从迭代结果看，随着迭代次数的增加，迭代值 $x^{(k)}$ 越来越接近精确解 $\begin{bmatrix} 1.1 \\ 1.2 \\ 1.3 \end{bmatrix}$

K	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0.00000	0.00000	0.00000
1	0.72000	0.83000	0.84000
2	0.97100	1.07000	1.15000
3	1.05700	1.15710	1.24820
4	1.08535	1.18534	1.28282
5	1.09510	1.19510	1.29414
6	1.09834	1.19834	1.29804
7	1.09944	1.19944	1.29934
8	1.09981	1.19981	1.29978
9	1.09994	1.19994	1.29992

设有**n** 阶方程组

$$
\begin{cases}
a_{11}x_1 + a_{12}x_2 + \Lambda + a_{1n}x_n = b_1 \\
a_{21}x_1 + a_{22}x_2 + \Lambda + a_{2n}x_n = b_2 \\
\quad \Lambda \qquad\quad \Lambda \qquad\qquad \Lambda \\
a_{n1}x_1 + a_{n2}x_2 + \Lambda + a_{nn}x_n = b_n
\end{cases}
\qquad (4)
$$

若系数矩阵非奇异，且 $a_{ii} \neq 0$ **($i = 1, 2, \ldots, n$)**，将上述方程组改写成

$$
\begin{cases}
x_1 = \dfrac{1}{a_{11}}\left(b_1 - a_{12}x_2 - a_{13}x_3 - \Lambda - a_{1n}x_n\right) \\
x_2 = \dfrac{1}{a_{22}}\left(b_2 - a_{22}x_1 - a_{23}x_3 - \Lambda - a_{2n}x_n\right) \\
\Lambda \ \Lambda \\
x_n = \dfrac{1}{a_{nn}}\left(b_{1n} - a_{n1}x_1 - a_{n2}x_2 - \Lambda - a_{n,n-1}x_{n-1}\right)
\end{cases}
$$

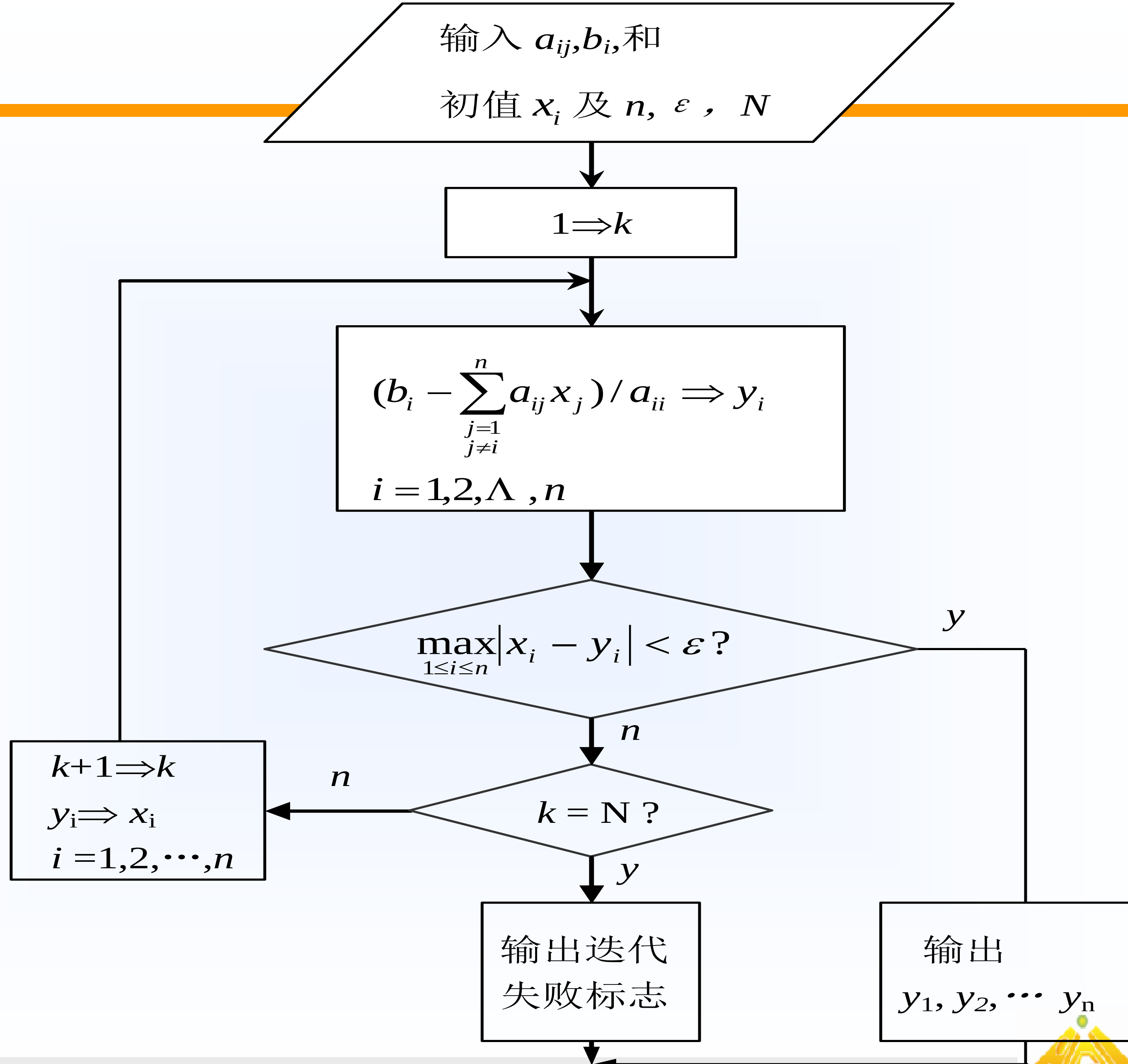
然后写成迭代格式

$$
\begin{cases}
x_1^{(k+1)} = \dfrac{1}{a_{11}}\left(b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \Lambda\ - a_{1n}x_n^{(k)}\right) \\
x_2^{(k+1)} = \dfrac{1}{a_{22}}\left(b_1 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \Lambda\ - a_{2n}x_n^{(k)}\right) \\
\Lambda\ \Lambda \\
x_n^{(k+1)} = \dfrac{1}{a_{nn}}\left(b_n - a_{1n}x_1^{(k)} - a_{n2}x_2^{(k)} - \Lambda\ - a_{n,n-1}x_{n-1}^{(k)}\right)
\end{cases}
$$

即得到解方程组的**雅可比迭代公式：**

$$
x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1,\, j\neq i}^{n} a_{ij}x_j^{(k)}\right) \qquad i = 1, 2, \mathrm{L}\ , n \qquad (5)
$$

雅可比迭代法的算法实现



2、高斯—塞德尔迭代

例：

$$\begin{cases} 10x_1 - x_2 - 2x_3 = 7.2 \\ -x_1 + 10x_2 - 2x_3 = 8.3 \\ -x_1 - x_2 + 5x_3 = 4.2 \end{cases}$$

解： 把x_1，x_2，x_3，从三个方程中分离出来得：

$$\begin{cases} x_1 = 0.1x_2 + 0.2x_3 + 0.72 \\ x_2 = 0.1x_1 + 0.2x_3 + 0.83 \\ x_3 = 0.2x_1 + 0.2x_3 + 0.84 \end{cases}$$

由此可建立雅可比迭代公式

$$
\begin{cases}
x_1^{(k+1)} = 0.1x_2^{(k)} + 0.2x_3^{(k)} + 0.72 & （1） \\
x_2^{(k+1)} = 0.1x_1^{(k)} + 0.2x_3^{(k)} + 0.83 & （2） \\
x_3^{(k+1)} = 0.2x_1^{(k)} + 0.2x_2^{(k)} + 0.84 & （3）
\end{cases}
$$

设已得到近似值 $(x_1^{(k)}, x_2^{(k)}, x_3^{(k)})$ 则由方程 (1) 得

$$x_1^{(k+1)} = 0.1x_2^{(k)} + 0.2x_3^{(k)} + 0.72$$

对于收敛的迭代过程， $x_1^{(k+1)}$一般比$x_1^{(k)}$更精确

所以，把 $x_1^{(k+1)}$ 代入 (2)

$$x_2^{(k+1)} = 0.1x_1^{(k+1)} + 0.2x_3^{(k)} + 0.83$$

则由此计算得到的计算结果 $x_2^{(k+1)}$ 要比用 $x_1^{(k)}$ 的好的多。

再把 $x_1^{(k+1)}$， $x_2^{(k+1)}$ 代入（3）

$$x_3^{(k+1)} = 0.2x_1^{(k+1)} + 0.2x_2^{(k+1)} + 0.84$$

取初值 $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$

按上述格式进行迭代计算得

K	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0.00000	0.00000	0.00000
1	0.72000	0.90200	1.16440
2	1.04308	1.16719	1.28205
3	1.09313	1.19572	1.29777
4	1.09913	1.19947	1.29972
5	1.09989	1.19993	1.29997
6	1.09999	1.19999	1.30000

而按雅可比迭代公式计算得

K	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0.00000	0.00000	0.00000
1	0.72000	0.83000	0.84000
2	0.97100	1.07000	1.15000
3	1.05700	1.15710	1.24820
4	1.08535	1.18534	1.28282
5	1.09510	1.19510	1.29414
6	1.09834	1.19834	1.29804
7	1.09944	1.19944	1.29934
8	1.09981	1.19981	1.29978
9	1.09994	1.19994	1.29992

刚才所述的方法称为高斯-塞德尔公式

对于一般形式的方程

$$x_1^{(k+1)} = \frac{1}{a_{11}}(-a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{14}x_4^{(k)} - \cdots - a_{1n}x_n^{(k)} + b_1)$$

$$x_2^{(k+1)} = \frac{1}{a_{22}}(-a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - a_{24}x_4^{(k)} - \cdots - a_{2n}x_n^{(k)} + b_2)$$

$$x_3^{(k+1)} = \frac{1}{a_{33}}(-a_{31}x_1^{(k+1)} - a_{32}x_2^{(k+1)} - a_{34}x_4^{(k)} - \cdots - a_{3n}x_n^{(k)} + b_3)$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots$$

$$x_n^{(k+1)} = \frac{1}{a_{nn}}(-a_{n1}x_1^{(k+1)} - a_{n2}x_2^{(k+1)} - a_{n3}x_3^{(k+1)} - \cdots - a_{nn-1}x_{n-1}^{(k+1)} + b_n)$$

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right) \quad i = 1,2,\cdots,n \qquad (7)$$

称为高斯—塞德尔迭代公式。

3、超松弛法

松弛法实质是高斯—塞德尔迭代的一种加速方法。

它将前一步的结果 $x_i^{(k)}$ 与高斯—塞德尔迭代值 $\tilde{x}_i^{(k+1)}$ 适当加权平均，以期望得到更好的近似值：

迭代 $$\tilde{x}_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right)$$

加速 $$x_i^{(k+1)} = \omega \tilde{x}_i^{(k+1)} + (1-\omega) x_i^{(k)} \qquad （8）$$

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right) \qquad （9）$$

超松弛迭代收敛的必要条件

式中系数 ω 称为松弛因子。

为保证迭代收敛，要求 $0<\omega<2$ 。

由于迭代值 $x_i^{(k+1)}$ 通常比 $x_i^{(k)}$ 精确，所以加大它的比重。

当 $0<\omega<1$ 该方法称为低松弛法

当 $\omega=1$ 该方法即为高斯-塞德尔迭代法

当 $1<\omega<2$

该方法称为超松弛（SOR Successive Over- Relaxation）法

例：

$$\begin{bmatrix} 5 & 1 & -1 & -2 \\ 2 & 8 & 1 & 3 \\ 1 & -2 & -4 & -1 \\ -1 & 3 & 2 & 7 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}=\begin{bmatrix} -2 \\ -6 \\ 6 \\ 12 \end{bmatrix}$$

试分别用**雅可比、高斯-塞德尔、超松弛法迭代**（取 $\omega=1.15$ ）解线性方程组。

解： 取 $x^{(0)}=(0,0,0,0)^T$

雅可比迭代公式为

$$\begin{cases} x_1^{(k+1)}=\dfrac{1}{5}(-2-x_2^{(k)}+x_3^{(k)}+2x_4^{(k)}) \\ x_2^{(k+1)}=\dfrac{1}{8}(-6-2x_1^{(k)}-x_3^{(k)}-3x_4^{(k)}) \\ x_3^{(k+1)}=-\dfrac{1}{4}(6-x_1^{(k)}+2x_2^{(k)}+x_4^{(k)}) \\ x_4^{(k+1)}=\dfrac{1}{7}(12+x_1^{(k)}-3x_2^{(k)}-2x_3^{(k)}) \end{cases}$$

迭代24次后得近似解

$$x^{(24)}=(0.9999941,-1.9999950,-1.0000040,2.9999990)$$

高斯-塞德尔迭代公式为

$$\begin{cases} x_1^{(k+1)} = \dfrac{1}{5}(-2 - x_2^{(k)} + x_3^{(k)} + 2x_4^{(k)}) \\ x_2^{(k+1)} = \dfrac{1}{8}(-6 - 2x_1^{(k+1)} - x_3^{(k)} - 3x_4^{(k)}) \\ x_3^{(k+1)} = -\dfrac{1}{4}(6 - x_1^{(k+1)} + 2x_2^{(k+1)} + x_4^{(k)}) \\ x_4^{(k+1)} = \dfrac{1}{7}(12 + x_1^{(k+1)} - 3x_2^{(k+1)} - 2x_3^{(k+1)}) \end{cases}$$

迭代9次后得近似解

$$x^{(9)} = (0.9999966, -1.9999970, -1.0000040, 2.9999990)$$

超松弛迭代公式为

$$\begin{cases} x_1^{(k+1)} = (1-\omega)x_1^{(k)} + \dfrac{\omega}{5}(-2 - x_2^{(k)} + x_3^{(k)} + 2x_4^{(k)}) \\ x_2^{(k+1)} = (1-\omega)x_2^{(k)} + \dfrac{\omega}{8}(-6 - 2x_1^{(k)} - x_3^{(k)} - 3x_4^{(k)}) \\ x_3^{(k+1)} = (1-\omega)x_3^{(k)} - \dfrac{\omega}{4}(6 - x_1^{(k)} + 2x_2^{(k)} + x_4^{(k)}) \\ x_4^{(k+1)} = (1-\omega)x_4^{(k)} + \dfrac{\omega}{7}(12 + x_1^{(k)} - 3x_2^{(k)} - 2x_3^{(k)}) \end{cases}$$

迭代8次后得近似解

$$x^{(8)} = (0.9999965, -1.9999970, -1.0000010, 2.9999990)$$

对于迭代法，指出两点

（1）从理论上讲，迭代法可以得到任意精度的近似解。但由于受机器字长的限制，不可能达到任意精度。故用误差估计式来控制迭代时，ε 要选取恰当，否则可能出现死循环。

（2）当给定的方程组不满足迭代的收敛条件时，可适当调整方程组中方程的次序或作一定的线性组合，就可能得到满足迭代收敛条件的同解方程组。

定义：n阶方阵A，如果其主对角线元素的绝对值大于同行其他元素的绝对值之和，则称A是严格行对角占优阵。

1.若A是严格对角占优矩阵，则对应的线性代数方程组有解。

2，如果A为严格对角占优矩阵，则A为非奇异矩阵。

3，若A为严格对角占优矩阵，则雅克比迭代法和高斯-赛德尔迭代法均收敛

例如：线性方程组

$$\begin{cases} 2x_1+9x_2=-5 \\ 8x_1+3x_2=13 \end{cases}$$

可以验证，无论用**雅可比迭代**或**高斯-塞德尔迭代**均不能满足迭代收敛条件。

但交换其方程的次序为

$$\begin{cases} 8x_1+3x_2=13 \\ 2x_1+9x_2=-5 \end{cases}$$

则该方程组的系数矩阵为严格对角占优。因此，无论用**雅可比迭代**或**高斯-塞德尔迭代**均满足迭代收敛条件。

例如： 线性方程组

$$\begin{cases} 5x_1+x_2+2x_3=0 & (1) \\ -11x_1+8x_2+x_3=21 & (2) \\ -4x_1-2x_2+3x_3=8 & (3) \end{cases}$$

（2）+（1）×2 ⟶ $-x_1+10x_2+5x_3=21$ （2）

（3）+（1） ⟶ $x_1-x_2+5x_3=8$ （3）

从而与原方程同解的线性方程组

$$\begin{cases} 5x_1+x_2+2x_3=0 \\ -x_1+10x_2+5x_3=21 \\ x_1 \quad -x_2 \quad +5x_3=8 \end{cases}$$

该方程组的系数矩阵为严格对角占优。因此，无论用**雅可比迭代**或**高斯-塞德尔迭代**均满足迭代收敛条件。

4、迭代公式的矩阵表示

线性方程组（4）可用矩阵记号简记为：

$$Ax = b \qquad (10)$$

式中

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

而

$$b = (b_1, b_2, \cdots, b_n)^T \qquad x = (x_1, x_2, \cdots, x_n)^T$$

设方程组 $Ax=b$ 的系数矩阵A非奇异，且主对角元素 $a_{ii}\neq 0(i=1,2,\cdots,n)$，则可将A分裂成

$$A=\begin{bmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ a_{31} & a_{32} & 0 & & \\ & & & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nn-1} & 0 \end{bmatrix}+\begin{bmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{bmatrix}+\begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ & 0 & a_{23} & \cdots & a_{2n} \\ & & 0 & & \\ & & & \ddots & a_{n-1n} \\ & & & & 0 \end{bmatrix}$$

记作　　$\mathbf{A=L+D+U}$

雅克比迭代公式的矩阵表示

则 $Ax=b$ 等价于 $(L+D+U)x=b$

伪对角形方程组

即 $Dx=-(L+U)x+b$

因为 $a_{ii}\neq 0(i=1,2,\Lambda\ ,n)$，则

$$x=-D^{-1}(L+U)x+D^{-1}b$$

则 得到一个迭代公式

Jacobi迭代的矩阵形式

$$x^{(k+1)}=-D^{-1}(L+U)x^{(k)}+D^{-1}b$$

$$=-D^{-1}(A-D)x^{(k)}+D^{-1}b$$

$$=(I-D^{-1}A)x^{(k)}+D^{-1}b$$

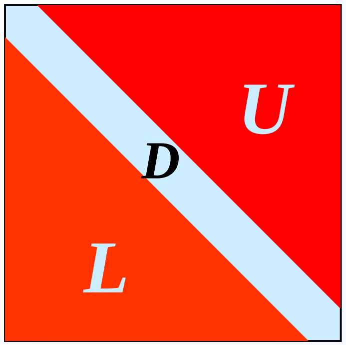
令 $G_1=-D^{-1}(L+U)\qquad d_1=D^{-1}b$

则有 $x^{(k+1)}=G_1x^{(k)}+d_1\qquad$ （k = 0,1,2…）

称为Jacobi迭代公式, G_1 称为Jacobi迭代矩阵

高斯—塞德尔迭代公式的矩阵表示

高斯—塞德尔迭代公式写成矩阵形式：

$A =$ 

$$\vec{x}^{(k+1)} = -D^{-1}(L\vec{x}^{(k+1)} + U\vec{x}^{(k)}) + D^{-1}\vec{b}$$

$$\Leftrightarrow \quad (D+L)\vec{x}^{(k+1)} = -U\vec{x}^{(k)} + \vec{b}$$

$$\Leftrightarrow \quad \vec{x}^{(k+1)} = \underbrace{-(D+L)^{-1}U}_{B}\vec{x}^{(k)} + \underbrace{(D+L)^{-1}\vec{b}}_{\vec{f}}$$

Gauss-Seidel
迭代阵

$$B = B_1 + B_2$$

$$B_1 = \begin{bmatrix} 0 & & & & \\ b_{21} & 0 & & & \\ b_{31} & b_{32} & 0 & & \\ \Lambda & & \Lambda & & \\ b_{n1} & b_{n2} & \Lambda & b_{nn-1} & 0 \end{bmatrix} \qquad B_2 = \begin{bmatrix} b_{11} & b_{12} & \Lambda & b_{1n-1} & b_{1n} \\ & b_{22} & \Lambda & b_{2n-1} & b_{2n} \\ & & \Lambda & \Lambda & \Lambda \\ \Lambda & & \Lambda & & \\ & & & & b_{nn} \end{bmatrix}$$

雅可比迭代公式 $x^{(k+1)} = Bx^{(k)} + f$

高斯-塞德尔公式 $x^{(k+1)} = B_1 x^{(k+1)} + B_2 x^{(k)} + f$

$$(I - B_1)x^{(k+1)} = B_2 x^{(k)} + f \Longrightarrow x^{(k+1)} = (I - B_1)^{-1} B_2 x^{(k)} + f$$

松弛法迭代公式的矩阵表示

$$x^{(k+1)} = (D-\omega L)^{-1}((1-\omega)D+\omega U)x^{(k)} + \omega(D-\omega L)^{-1}b$$

令 $B_\omega = (D-\omega L)^{-1}((1-\omega)D+\omega U)$

$$f_\omega = \omega(D-\omega L)^{-1}b$$

$$x^{(k+1)} = B_\omega x^{(k)} + f_\omega$$

例：写出下列方程组的G-S迭代格式

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

$$\Longrightarrow \begin{cases} x_1^{(k+1)} = \frac{1}{8}\left[20 + 3x_2^{(k)} - 2x_3^{(k)}\right] \\ x_2^{(k+1)} = \frac{1}{11}\left[33 - 4x_1^{(k)} + x_3^{(k)}\right] \\ x_3^{(k+1)} = \frac{1}{4}\left[12 - 2x_1^{(k)} - x_2^{(k)}\right] \end{cases}$$

$$\begin{cases} x_1^{(k+1)} = \frac{1}{8}\left[20 + 3x_2^{(k)} - 2x_3^{(k)}\right] \\ x_2^{(k+1)} = \frac{1}{11}\left[33 - 4x_1^{(k+1)} + x_3^{(k)}\right] \\ x_3^{(k+1)} = \frac{1}{4}\left[12 - 2x_1^{(k+1)} - x_2^{(k+1)}\right] \end{cases}$$

习题

Ｐ１７０　１、3、５、７