电路计算机辅助分析基础

叶庆云　编

武汉工业大学出版社

电路计算机辅助分析基础

叶庆云　编

武汉工业大学出版社

内容简介

本书比较详细地讨论了电路的计算机辅助分析的基本理论、方法和程序实现，并适当地引入了新近发展的双图改进节点法。全书共四章，内容包括：电路元器件模型，电路方程组的自动建立，线性代数方程组的数值解，线性电路的稳态分析，线性电路的瞬态分析，非线性电路的直流分析。书中内容丰富、条理清楚、结合实例、深入浅出。在各章中都给出了相应的分析计算程序的较详细的框图，并对程序编写的有关问题作了详细的介绍。

本书可作为高等院校电子工程、自动控制和工业自动化等专业的必修课或选修课的教材及有关专业师生的教学参考书。

图书在版编目(CIP)数据

电路计算机辅助分析基础/叶庆云编. —武汉:武汉工业大学出版社,1999.9
ISBN 7-5629-1519-9
Ⅰ.电… Ⅱ.叶… Ⅲ.电路-计算机辅助分析 Ⅳ.TM13

中国版本图书馆 CIP 数据核字(99)第 38945 号

武汉工业大学出版社出版发行
(武汉市武昌珞狮路 122 号　邮政编码 430070)
各地新华书店经销
武汉工业大学出版社印刷
*
开本:787×1092　1/16　印张:9.5　字数:240 千字
1999 年 9 月第 1 版　1999 年 9 月第 1 次印刷
印数:1～2000 册
定价:15.00 元

前　言

电路CAD技术——电路的计算机辅助分析(Computer Aided Analysis 简称CAA)和计算机辅助设计(Computer Aided Design 简称CAD),是在计算数学、电路理论和程序设计基础上发展起来的一门独特的新技术学科。它把电子计算机的快速、高精度、存储容量大,严格的逻辑判断和优良的数据处理能力与人的创造思维能力充分结合起来,为现代电路设计开辟了广阔的前景,成为当今电路设计工作者必须掌握的一门重要技术。

本书作为电路分析与设计的计算机化的入门,着重介绍电路的计算机辅助分析的基本原理、实用算法和程序实现。为了使学生不仅能较好地掌握基本原理和算法,还能自己编写程序,书中对各章程序的编写原理和方法给予了足够的重视,对一些通用的程序列有较详细的框图,并配有实例供学生上机练习,便于学生学以致用。

为了掌握本书内容,读者应具备线性代数、计算数学、电路理论、QBASIC 或 FORTRAN 语言及程序设计等方面的基础知识。对于理工科大学的本科高年级,讲授本书内容需50学时左右。

本书是作者在总结多年来科研和教学实践经验的基础上,参考了国内外的大量有关文献资料,经过系统整理,删繁增新编写而成。它力图体现知识的系统性、理论性和应用性,做到算法和编程相结合。

本书在编写过程中得到武汉工业大学教务处的大力支持,娄桂泉、周省三两位教授在百忙之中详细审阅了全部原稿,并对书稿提出了许多极为有益的建议和详细的修改意见,在此一并致以深切谢意。同时对所有为本书出版作出贡献的同志们,对本书在编写过程中所有参考过的文献资料的作者们,一并表示衷心感谢。

由于编者水平有限,经验不足,书中难免存在不少错误和不妥之处,敬请读者批评指正。

编　者

于武汉工业大学

1998.12

目　录

第一章　线性代数方程组的数值解

线性代数方程组的数值解法是电路的计算机辅助分析的数学基础之一。这是因为线性电路的稳态分析，最后归结为求解实系数或复系数的线性代数方程组，它的一般形式为：

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

或缩写为循环体形式

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \quad i = 1,2,\cdots,n \tag{1-1-2}$$

或写成矩阵形式

$$\boldsymbol{AX} = \boldsymbol{B} \tag{1-1-3}$$

式中 A 是方程组的系数矩阵，$\boldsymbol{X}$ 是未知量列向量，$\boldsymbol{B}$ 是右端列向量，即

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \qquad \boldsymbol{X} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \boldsymbol{B} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \tag{1-1-4}$$

对(1-1-1)式进行物理分析可知，系数矩阵 A 总具有对角线优势(即轴元素优势)，且轴元素不会为零，非轴元素为零者常见，即系数矩阵 A 具有稀疏现象。例如，在用节点法分析电路时，必须求解节点电压方程 $Y_n\boldsymbol{V}_n=\boldsymbol{J}_n$，此时节点导纳矩阵 Y_n 的轴元素是某节点的自导，而非轴元素则为两节点间的互导，所以 Y_n 的轴元素都是非零元(以后简写为 NZ)，而非轴元素是否为零，则取决于两节点间是否有某些支路相联系。事实上，在实际电路里有不少节点间并不存在支路或耦合关系，所以非轴元素中有很多零元(以后简写为 Z)。据统计，电力系统中，节点导纳矩阵内每行每列的 NZ 元平均不超过四个，因此 Y_n 内 NZ 的个数，大致与节点数成正比，系统越大 Y_n 的稀疏程度越高。电子线路中情况类似，当用改进节点法建立电路方程时，其系数矩阵的每行每列 NZ 的个数平均为 4～6 个，因此这种矩阵也是极其稀疏的。我们称之为“稀疏矩阵”，与之对应的线性方程组则称之为稀疏线性方程组。为定量描述矩阵的稀疏法，定义矩阵内零元 Z 占全部元素的百分比为稀疏率：

$$\text{稀疏率 } S = \text{零元数目 } N_Z / \text{ 总元数目 } N_T \tag{1-1-5}$$

在大规模电路及其他工程系统的分析中，稀疏率可达到 95%。

求解稀疏线性方程组通常有两类解法，即迭代法和直接法。

迭代法是给定初值，通过逐次逼近来解线性代数方程组，其迭代过程一直进行到近似解达到预定的精度为止，如高斯-塞德尔迭代法。迭代法的优点是只有矩阵中的非零元参加运算，只需存贮原方程中的 NZ 元，因此存贮单元最节省，但由于迭代法对系数矩阵的要求往往比较苛刻，否则，其收敛性及收敛速度都难于保证。因此，近 20 年来，除了使用迭代法的比较传统的领域外，人们往往更倾向于使用直接法。

本章将采用直接法，它是通过对方程组直接进行有限步初等数学运算，得到方程的解。其运算步骤一定，运算次数有限，解的精度取决于矩阵的性质和所用的算法及计算机的字长等。这里主要介绍 Gauss 消去法、LU 分解法以及程序实现。

第一节　高斯(Gauss)消去法

下面以一个三阶的方程组为实例说明高斯消去法的计算过程。

$$3x_1 + 2x_2 + x_3 = 6 \tag{1-1-6a}$$

$$2x_1 + 2x_2 + 2x_3 = 4 \tag{1-1-6b}$$

$$4x_1 - 2x_2 - 2x_3 = 2 \tag{1-1-6c}$$

第一步：目的是使(1-1-6a)式 x_1 系数化为 1，(1-1-6b)和(1-1-6c)式 x_1 系数化为零。

$$(1\text{-}1\text{-}6a) \times \frac{1}{3}：1x_1 + \frac{2}{3}x_2 + \frac{1}{3}x_3 = 2 \tag*{(1-1-6a)$'$}$$

$$(1\text{-}1\text{-}6a)' \times (-2) + (1\text{-}1\text{-}6b)：0x_1 + \frac{2}{3}x_2 + \frac{4}{3}x_3 = 0 \tag*{(1-1-6b)$'$}$$

$$(1\text{-}1\text{-}6a)' \times (-4) + (1\text{-}1\text{-}6c)：0x_1 - \frac{14}{3}x_2 - \frac{10}{3}x_3 = -6 \tag*{(1-1-6c)$'$}$$

第二步：目的是保留方程(1-1-6a)$'$，且使(1-1-6b)$'$式 x_2 系数化为 1，(1-1-6c)$'$式 x_2 系数化为零。

$$1x_1 + \frac{2}{3}x_2 + \frac{1}{3}x_3 = 2 \tag*{(1-1-6a)$'$}$$

$$(1\text{-}1\text{-}6b)' \times \frac{3}{2}：0x_1 + 1x_2 + 2x_3 = 0 \tag*{(1-1-6b)$''$}$$

$$(1\text{-}1\text{-}6b)'' \times \frac{14}{3} + (1\text{-}1\text{-}6c)'：0x_1 + 0x_2 + 6x_3 = -6 \tag*{(1-1-6c)$''$}$$

第三步：目的是保留方程(1-1-6a)$'$、(1-1-6b)$''$，且使(1-1-6c)$''$式 x_3 系数化为 1。

$$1x_1 + \frac{2}{3}x_2 + \frac{1}{3}x_3 = 2 \tag*{(1-1-6a)$'$}$$

$$0x_1 + 1x_2 + 2x_3 = 0 \tag*{(1-1-6b)$''$}$$

$$(1\text{-}1\text{-}6c)'' \times \frac{1}{6}：0x_1 + 0x_2 + 1x_3 = -1 \tag*{(1-1-6c)$'''$}$$

以上三步称为正向消去过程。

用矩阵表示：$\boldsymbol{AX}=\boldsymbol{B} \xrightarrow{\text{经过三步变换}} \boldsymbol{A}^{*}\boldsymbol{X}=\boldsymbol{B}^{*}$

具体过程为

$$\begin{bmatrix} 3 & 2 & 1 \\ 2 & 2 & 2 \\ 4 & -2 & -2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \\ 2 \end{bmatrix} \xrightarrow[\text{三次行变换}]{\text{第一步}} \begin{bmatrix} 1 & \frac{2}{3} & \frac{1}{3} \\ 0 & \frac{2}{3} & \frac{4}{3} \\ 0 & -\frac{14}{3} & -\frac{10}{3} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ -6 \end{bmatrix}$$

$$\xrightarrow[\text{二次行变换}]{\text{第二步}}\begin{bmatrix}1 & \frac{2}{3} & \frac{1}{3}\\0 & 1 & 2\\0 & 0 & 6\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}2\\0\\-6\end{bmatrix}\xrightarrow[\text{一次行变换}]{\text{第三步}}\begin{bmatrix}1 & \frac{2}{3} & \frac{1}{3}\\0 & 1 & 2\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}2\\0\\-1\end{bmatrix}$$

根据第三步变换后所得到的方程组，我们很容易通过反向回代求出解答。

$$\begin{aligned}x_3&=-1\\x_2&=0-2x_3=0-2(-1)=2\\x_1&=2-\frac{1}{3}x_3-\frac{2}{3}x_2=2-\frac{1}{3}(-1)-\frac{2}{3}(2)=1\end{aligned}\tag{1-1-7}$$

通过上述例子，我们对高斯消去法的基本思想有了了解，这就是高斯消去法包括两个过程，即正向消去和反向回代过程。在消去过程中的每一步，首先用轴元除它所在方程各未知数的系数和右端项，称为轴元归一化①，然后依次留下轴元归一后的方程，逐个地消去其余方程中的未知数，直到剩下最后一个方程仅有一个未知数 x_3 为止。其实质是经过一系列初等行变换将方程组的系数矩阵 A 变换成等效的上三角形矩阵 $U(A^*)$，右端列向量相应变换为 $\boldsymbol{B}^*$。然后执行反向回代，首先从最后一个方程中解出 x_3，将 x_3 代入第二个方程，求出 x_2，以此类推，直到求出 x_1 为止。

现在我们将此法推广到 n 阶线性方程组，得出一般计算公式。

设一 n 阶线性方程组 $A\boldsymbol{X}=\boldsymbol{B}$，为方便起见，我们用矩阵 A 和列向量 $\boldsymbol{B}$ 组成的 $n\times(n+1)$ 维的增广矩阵用 A_a 来表示，且右端项作为第 $(n+1)$ 列统一编号，即

$$A_a=\left[\begin{array}{cccc:c}a_{11} & a_{12} & \cdots & a_{1n} & a_{1,n+1}\\a_{21} & a_{22} & \cdots & a_{2n} & a_{2,n+1}\\\cdots & \cdots & \cdots & \cdots & \cdots\\a_{i1} & a_{i2} & \cdots & a_{in} & a_{i,n+1}\\\cdots & \cdots & \cdots & \cdots & \cdots\\a_{n1} & a_{n2} & \cdots & a_{nn} & a_{n,n+1}\end{array}\right]\tag{1-1-8}$$

在消去过程的第一步，首先用 a_{11} 除 A_a 中第一行所有元素，使之归一化，然后对下面 $(n-1)$ 行中第 i 行分别用 $(-a_{i1})$ 乘以新的第一行再与第 i 行相加，以消去第一列中的其余元素，此时 A_a 变为

$$A_a^{(1)}=\left[\begin{array}{c:ccccc}1 & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} & a_{1,n+1}^{(1)}\\\hdashline 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} & a_{2,n+1}^{(1)}\\\cdots & \cdots & \cdots & \cdots & \cdots & \cdots\\\cdots & a_{i2}^{(1)} & a_{i3}^{(1)} & \cdots & a_{in}^{(1)} & a_{i,n+1}^{(1)}\\\cdots & \cdots & \cdots & \cdots & \cdots & \cdots\\0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \cdots & a_{nn}^{(1)} & a_{n,n+1}^{(1)}\end{array}\right]\tag{1-1-9}$$

其中 $a_{1j}^{(1)}=a_{1j}/a_{11},\quad j=1,2,\cdots,n+1$

$a_{ij}^{(1)}=a_{ij}-a_{i1}\cdot a_{1j}^{(1)},\quad i=2,3,\cdots,n;j=1,2,\cdots,n+1$

在消去过程的第二步，对 $A_a^{(1)}$ 的剩余矩阵中，首先用 $a_{22}^{(1)}$ 除以第二行所有元素，使之归一化，然后对下面 $(n-2)$ 行中第 i 行分别用 $(-a_{i2}^{(1)})$ 乘以新的第二行再与第 i 行相加，以消去第二

①其目的是防止经过每次消元运算的矩阵各元素值变大。

列中的其余元素，此时 $A_a^{(1)}$ 变为

$$A_a^{(2)} = \left[\begin{array}{cc:ccc:c} 1 & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} & a_{1,n+1}^{(1)} \\ 0 & 1 & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} & a_{2,n+1}^{(2)} \\ \hdashline \cdots & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} & a_{3,n+1}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} & a_{n,n+1}^{(2)} \end{array}\right] \tag{1-1-10}$$

其中 $a_{2j}^{(2)}=a_{2j}^{(1)}/a_{22}^{(1)}, \quad j=2,3,\cdots,n+1$

$a_{ij}^{(2)}=a_{ij}^{(1)}-a_{i2}^{(1)}\cdot a_{2j}^{(2)}, \quad i=3,4,\cdots,n;j=2,3,\cdots,n+1$

以此类推，一直进行到第 n 步，第 n 步的结果为

$$A_a^{(n)} = \left[\begin{array}{ccccccc:c} 1 & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & \cdots & \cdots & a_{1n}^{(1)} & a_{1,n+1}^{(1)} \\ 0 & 1 & a_{23}^{(2)} & \cdots & \cdots & \cdots & a_{2n}^{(2)} & a_{2,n+1}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & 1 & a_{k,k+1}^{(k)} & \cdots & a_{kn}^{(k)} & a_{k,n+1}^{(k)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & \cdots & 0 & 1 & a_{n,n+1}^{(n)} \end{array}\right] \tag{1-1-11}$$

其中第 k 步，是对矩阵 $A_a^{(k-1)}$ 的剩余矩阵进行下列运算：

$a_{kj}^{(k)}=a_{kj}^{(k-1)}/a_{k}^{(k-1)}, \quad j=k,k+1,\cdots,n+1$

$a_{ij}^{(k)}=a_{ij}^{(k-1)}-a_{ik}^{(k-1)}\cdot a_{kj}^{(k)}, \quad i=k+1,k+2,\cdots,n;j=k,k+1,\cdots,n+1$

将增广矩阵 $A_a^{(n)}$ 还原成线性代数方程组如下：

$$\begin{aligned} x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1n}^{(1)}x_n &= a_{1,n+1}^{(1)} \\ x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2n}^{(2)}x_n &= a_{2,n+1}^{(2)} \\ \cdots\cdots & \\ x_n &= a_{n,n+1}^{(n)} \end{aligned} \tag{1-1-12}$$

该方程组的特点是后一个方程较前一个方程依次少一个未知数，而最后一个方程仅是一元一次方程，显然依次向上回代，就可以得出全部解序列 $x_n,x_{n-1},\cdots,x_2,x_1$。

综上所述高斯消去法的整个过程计算公式如下：

1. 正向消去公式

$$\begin{aligned} a_{kj}^{(k)} &= a_{kj}^{(k-1)}/a_{kk}^{(k-1)} \\ a_{ij}^{(k)} &= a_{ij}^{(k-1)} - a_{ik}^{(k-1)}\cdot a_{kj}^{(k)} \end{aligned} \qquad \begin{cases} k = 1,2,\cdots,n \\ j = k,k+1,\cdots,n+1 \\ i = k+1,k+2,\cdots,n \end{cases} \tag{1-1-13}$$

2. 反向回代公式

$$x_n = a_{n,n+1}^{(n)}$$

$$x_i = a_{i,n+1}^{(i)} - \sum_{j=i+1}^{n} a_{ij}^{(i)}x_j, \quad i = n-1,n-2,\cdots,1 \tag{1-1-14}$$

上述高斯消去法，在实际使用中存在两个问题。

1. 以上按自然顺序消去过程得以进行到底，是假设 $a_{kk}^{(k-1)}\neq 0(k=1,\cdots,n)$ 均成立的缘故。如果出现 $a_{kk}^{(k-1)}=0$ 的情况，即使原方程存在唯一解，上述消去过程亦将无法进行。例如，对于方程组

$$\begin{cases} 0x_1 + x_2 = 1 \\ x_1 + x_2 = 2 \end{cases}$$

很显然，它的解是 $x_1=x_2=1$。但若机械地采用上述自然顺序消元过程，则在第一步就无法进行。如果将方程组改为

$$\begin{cases} x_1 + x_2 = 2 \\ 0x_1 + x_2 = 1 \end{cases}$$

就能解决这个问题，且对解没有影响。

2. 即使 $a_{kk}^{(k-1)}\neq 0(k=1,\cdots,n)$ 总成立，但当 $|a_{kk}^{(k-1)}|$ 相对于同一列中下面其余元素的绝对值 $|a_{ik}^{(k-1)}|(i>k)$ 为小值时，虽然从理论上说，消去过程是可以进行的，但往往会在下一步的消元中引入很大的舍入误差，经验表明，一般说来 $a_{11},a_{22}^{(1)},\cdots,a_{kk}^{(k-1)},\cdots$ 作为消元时的除数，其绝对值越小，舍入误差也就越严重，由于这些误差的传播和积累而使计算精度降低，甚至导致错误的后果。

基于上述二方面考虑，人们提出了各种选主元素的方法，使得只要 A 阵非奇异，消元过程便不致中断，另一方面又可减少误差积累，提高精度。

选主元素的基本思想是，在消去过程中，必须选择方程中系数的绝对值最大的那个未知量作为消去元，称为主元素，同时，把主元素所在的方程作为保留方程。把这个原则应用到 n 阶线性代数方程组的求解过程中去，就称为主元消去法。其中又有列、行、全主元消去法之分。

列主元消去法是在 $A_a^{(k-1)}$ 的剩余矩阵中，选本列（k 列）的绝对值最大者作为主元，其主元应使如下关系式成立

$$|a_{k'k}^{(k-1)}| = \max_i |a_{ik}^{(k-1)}| \quad (i \geqslant k)$$

即在进行第 k 步消元时，先对 k 列中第 k 行至第 n 行中元素选出绝对值最大值为主元，如 $|a_{k'k}^{(k-1)}|$ 为最大，则将第 k' 行与第 k 行对调（行交换）。然后再进行主元归一和消元。

$$A_a^{(k-1)} = \left[\begin{array}{ccc:cccc:c} 1 & a_{12}^{(1)} & \cdots & \cdots & \cdots & \cdots & a_{1,n+1}^{(1)} \\ & 1 & a_{23}^{(2)} & \cdots & \cdots & \cdots & a_{2,n+1}^{(2)} \\ & & \cdots & \cdots & \cdots & \cdots & \vdots \\ \hdashline & & \downarrow & a_{kk}^{(k-1)} & \cdots & \cdots & a_{k,n+1}^{(k-1)} \\ & & i\ \vdots & a_{k'k}^{(k-1)} & \downarrow\uparrow & \text{对调} & a_{k',n+1}^{(k-1)} \\ & & \downarrow & a_{nk}^{(k-1)} & \cdots & a_{nn}^{(k-1)} & a_{n,n+1}^{(k-1)} \end{array}\right]$$

行主元消去法是在 $A_a^{(k-1)}$ 的剩余矩阵中选本行（k 行）的绝对值最大者作为主元，其主元应使如下关系式成立

$$|a_{kk'}^{(k-1)}| = \max_j |a_{kj}^{(k-1)}| \quad (j \geqslant k)$$

即在进行第 k 步消元时，先对 k 行中第 k 列至第 n 列中元素选出绝对值最大者为主元，如 $|a_{kk'}^{(k-1)}|$ 为最大，则将第 k' 列与第 k 列对调（列交换）。然后再进行主元归一和消元。

$$A_a^{(k-1)} = \left[\begin{array}{ccc:ccccc:c} 1 & a_{12}^{(1)} & \cdots & \cdots & \cdots & \cdots & \cdots & a_{1,n+1}^{(1)} \\ & 1 & a_{23}^{(2)} & \cdots & \cdots & \cdots & \cdots & a_{2,n+1}^{(2)} \\ & & & \rightarrow & \overset{j}{\cdots} & \rightarrow & & \\ \hdashline & & & a_{kk}^{(k-1)} & \cdots & a_{kk'}^{(k-1)} & \cdots & a_{k,n+1}^{(k-1)} \\ & & & \vdots & \overset{\text{对调}}{\leftrightarrows} & \vdots & \cdots & \vdots \\ & & & a_{nk}^{(k-1)} & \cdots & a_{nk'}^{(k-1)} & \cdots & a_{n,n+1}^{(k-1)} \end{array}\right]$$

全主元消去法是在 $A_a^{(k-1)}$ 的剩余矩阵中选本行本列（k 行 k 列）的绝对值最大者作为主

元，其主元应使如下关系式成立。

$$|a_{k'k''}^{(k-1)}| = \max_{ij}|a_{ij}^{(k-1)}| \quad (i,j \geqslant k)$$

即在进行第 k 步消元时，选出主元 $|a_{k'k''}^{(k-1)}|$，并将第 k' 行与第 k 行，第 k'' 列与第 k 列进行对调，使 $|a_{k'k''}^{(k-1)}|$ 交换到主元位置上，然后再进行第 k 步的主元归一和消元。

$$A_a^{(k-1)} = \left[\begin{array}{ccc:cccc:c} 1 & a_{12}^{(1)} & \cdots & \cdots & \cdots & \cdots & \cdots & a_{1,n+1}^{(1)} \\ & 1 & a_{23}^{(2)} & \cdots & \cdots & \cdots & \cdots & a_{2,n+1}^{(2)} \\ & & & \rightarrow & \overset{j}{\cdots} & \rightarrow & & \\ \hdashline & & \downarrow & a_{kk}^{(k-1)} & \cdots & a_{kk''}^{(k-1)} & \cdots & a_{k,n+1}^{(k-1)} \\ & & i\ \vdots & a_{k'k}^{(k-1)} & \text{对调} & a_{k'k''}^{(k-1)} & \cdots & \vdots \\ & & \downarrow & a_{nk}^{(k-1)} & \cdots & a_{nk''}^{(k-1)} & \cdots & a_{n,n+1}^{(k-1)} \end{array}\right]$$

三种方法比较一下，就计算精度来说，全主元消去法效果更好些，但工作量也更大，对大多数实际问题来说，列主元消去法已能满足精度要求，且不必变更未知数的顺序，具有编程简单的优点，故普遍采用列主元高斯消去法。

第二节　列主元高斯消去法及程序的编写

前面介绍高斯消去法的求解过程时，已经得到如(1-1-13)式和(1-1-14)式的消去和回代过程的计算公式。它们是高斯消去法编程的依据。列主元高斯消去法除在每一步需增加按列选主元并进行行交换外，其余步骤与高斯消去法相同。

对列主元高斯消去法程序的编写，作如下两点说明。

1. 求解过程及计算公式

$$\left.\begin{array}{l} \text{(1)按列选主元并换行} \\ \quad |a_{k'k}^{(k-1)}| = \max\limits_{i}|a_{ik}^{(k-1)}| \quad (i \geqslant k, k'\text{行与 }k\text{ 行交换}) \\ \text{(2)主元归一及消元} \\ \quad \begin{array}{ll} a_{kj}^{(k)} = a_{kj}^{(k-1)}/a_{kk}^{(k-1)} & \left(\begin{array}{l} j=k,k+1,\cdots,n+1 \\ i=k+1,k+2,\cdots,n \end{array}\right) \\ a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} \cdot a_{kj}^{(k)} & \end{array} \end{array}\right\} (k=1,2,\cdots,n) \tag{1-2-1}$$

(3)反向回代

$$\begin{aligned} x_n &= a_{n,n+1}^{(n)} \\ x_i &= a_{i,n+1}^{(i)} - \sum_{j=i+1}^{n} a_{ij}^{(i)} x_j \end{aligned} \quad (i = n-1, n-2, \cdots, 1) \tag{1-2-2}$$

在编程时，应尽量减少不必要的运算，以节约机时．由于考虑到消去过程结束时，$A_a^{(n)}$ 中只有对角线以上的元素参与回代运算，所以(1-2-1)中 $j=k$ 这一步实际上不需要进行，而且 $x_n = a_{n,n+1}^{(n)}$ 是现成的也无需计算，故在编程时应将公式修改如下

$$\left.\begin{array}{l} \text{(1)按列选主元并换行} \\ \quad |a_{k'k}^{(k-1)}| = \max\limits_{i}|a_{ik}^{(k-1)}| \quad (i \geqslant k, k'\text{行与 }k\text{ 行交换}) \\ \text{(2)主元归一及消元} \\ \quad \begin{array}{ll} a_{kj}^{(k)} = a_{kj}^{(k-1)}/a_{kk}^{(k-1)} & \left(\begin{array}{l} j=k+1,k+2,\cdots,n+1 \\ i=k+1,k+2,\cdots,n \end{array}\right) \\ a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} \cdot a_{kj}^{(k)} & \end{array} \end{array}\right\} (k=1,2,\cdots,n) \tag{1-2-3}$$

(3)反向回代

$$x_i = a_{i,n+1}^{(i)} - \sum_{j=i+1}^{n} a_{ij}^{(i)} x_j \quad (i = n-1, n-2, \cdots, 1) \tag{1-2-4}$$

2. 存贮空间

列主元高斯消去法的整个运算过程是在一个 $n \times (n+1)$维的增广矩阵 A_a 内进行的，在计算机中需占用一个二维数组的存贮空间，即 $A(N,N+1)$。在程序运行前，A 数组中前 N 列存放系数矩阵 A，第 $N+1$ 列存放右端列向量 $\boldsymbol{B}$；程序运行过程中，当用主元归一公式由 $a_{kj}^{(k-1)}$ 计算出 $a_{kj}^{(k)}$之后，原 $a_{kj}^{(k-1)}$不再参与以后的运算，因此可让新的 $a_{kj}^{(k)}$ 占用原 $a_{kj}^{(k-1)}$的存贮单元；同理 $a_{ij}^{(k)}$ 占用 $a_{ij}^{(k-1)}$的存贮单元，即不断用新值冲旧值；在计算 x_i 时，每当计算完毕一个 x_i，前 $a_{i,n+1}^{(i)}$ 同样不再参与今后的运算，亦可让 x_i 占用 $a_{i,n+1}^{(i)}$的存贮单元。这样，求解的整个过程自始至终在一个二维数组中进行。最后计算结果就存放在原来存放右端列向量的第 $N+1$ 列中。

EPS 是一个作控制常数用的实变量。当主元绝对值趋于零时，意味着系数矩阵是奇异的，考虑到计算机舍入等方面的误差，设立一个小的正实数(如 1E—35)，若主元绝对值小于 ESP，则停止运算。

按照以上三点说明，设计出的列主元高斯消去法程序框图见图 1-2-1。

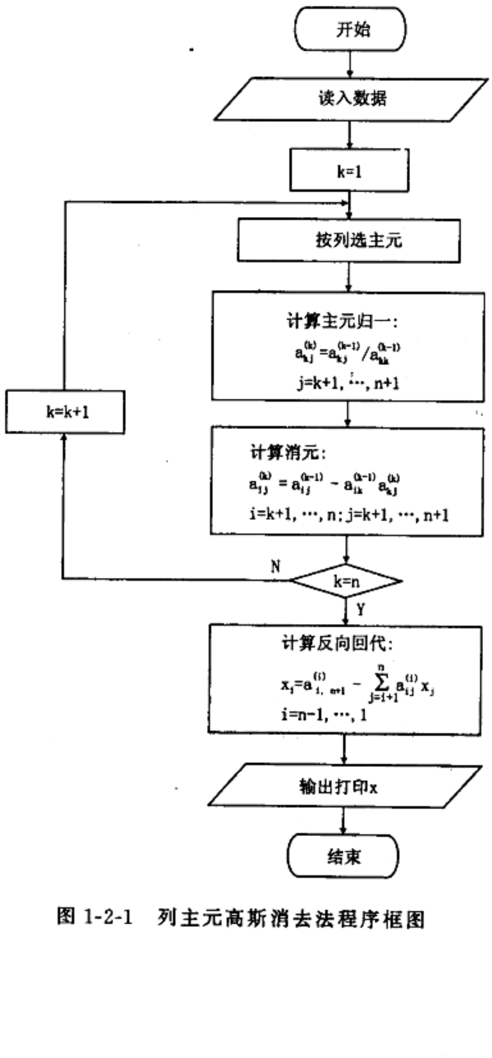


图 1-2-1　列主元高斯消去法程序框图

第三节 LU分解法——Crout分解法

对于给定的线性代数方程组

$$AX = B \tag{1-3-1}$$

利用LU分解法可以将系数矩阵A分解成如下的两个三角形矩阵的乘积，即使

$$A = LU \tag{1-3-2}$$

式中L为下三角形矩阵对角线上方元素为零

$$L = \begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \tag{1-3-3}$$

U是一个主对角线元素为1的上三角形矩阵

$$U = \begin{bmatrix} 1 & U_{12} & \cdots & U_{1n} \\ 0 & 1 & & \cdots \\ \cdots & \cdots & \cdots & U_{n-1,n} \\ 0 & \cdots & \cdots & 1 \end{bmatrix} \tag{1-3-4}$$

将(1-3-2)式代入(1-3-1)式，得

$$LUX = B \tag{1-3-5}$$

令

$$UX = Y \tag{1-3-6}$$

则

$$LY = B \tag{1-3-7}$$

由于L,U均为三角形矩阵，如果先根据(1-3-7)式利用正向代入求出Y，然后代入(1-3-6)式，利用反向回代，则很方便地求出X，即原方程组的解。这种方法实际上是高斯消去法的一种变形。只是它更直观简便罢了。

现在的问题是如何对A进行分解，我们以一个四阶方程组为例说明其计算步骤。然后再给出它的计算通式。

设方程组为

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

首先把A分解为L和U的乘积，并利用矩阵乘法求出L和U中的各对应元素(这是假设这些因子均存在)。则可写出

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

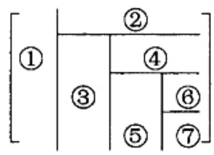
这时

$$L \cdot U = \begin{bmatrix} l_{11} & l_{11}u_{12} & l_{11}u_{13} & l_{11}u_{14} \\ l_{21} & l_{21}u_{12}+l_{22} & l_{21}u_{13}+l_{22}u_{23} & l_{21}u_{14}+l_{22}u_{24} \\ l_{31} & l_{31}u_{12}+l_{32} & l_{31}u_{13}+l_{32}u_{23}+l_{33} & l_{31}u_{14}+l_{32}u_{24}+l_{33}u_{34} \\ l_{41} & l_{41}u_{12}+l_{42} & l_{41}u_{13}+l_{42}u_{23}+l_{43} & l_{41}u_{14}+l_{42}u_{24}+l_{43}u_{34}+l_{44} \end{bmatrix}$$

通常设置一个辅助矩阵 Q,其中存放 L 和 U 中除 1 以外的元素

$$Q = L + (U - I) = \begin{bmatrix} l_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & l_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & l_{33} & u_{34} \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix}$$

与上述分解过程对应 Q 中各元素的计算顺序按先列后行,依次交替进行,其编号顺序为



这样就有下述关系式

$$①\begin{cases} l_{11}=a_{11} \\ l_{21}=a_{21} \\ l_{31}=a_{31} \\ l_{41}=a_{41} \end{cases}$$

$$②\begin{cases} l_{11}u_{12}=a_{12} \\ l_{11}u_{13}=a_{13} \\ l_{11}u_{14}=a_{14} \end{cases} \qquad \begin{cases} u_{12}=a_{12}/l_{11} \\ u_{13}=a_{13}/l_{11} \\ u_{14}=a_{14}/l_{11} \end{cases}$$

$$③\begin{cases} l_{21}u_{12}+l_{22}=a_{22} \\ l_{31}u_{12}+l_{32}=a_{32} \\ l_{41}u_{12}+l_{42}=a_{42} \end{cases} \qquad \begin{cases} l_{22}=a_{22}-l_{21}u_{12} \\ l_{32}=a_{32}-l_{31}u_{12} \\ l_{42}=a_{42}-l_{41}u_{12} \end{cases}$$

$$④\begin{cases} l_{21}u_{13}+l_{22}u_{23}=a_{23} \\ l_{21}u_{14}+l_{22}u_{24}=a_{24} \end{cases} \qquad \begin{cases} u_{23}=(a_{23}-l_{21}u_{13})/l_{22} \\ u_{24}=(a_{24}-l_{21}u_{14})/l_{22} \end{cases}$$

$$⑤\begin{cases} l_{31}u_{13}+l_{32}u_{23}+l_{33}=a_{33} \\ l_{41}u_{13}+l_{42}u_{23}+l_{43}=a_{43} \end{cases} \qquad \begin{cases} l_{33}=a_{33}-l_{31}u_{13}-l_{32}u_{23} \\ l_{43}=a_{43}-l_{41}u_{13}-l_{42}u_{23} \end{cases}$$

$$⑥ l_{31}u_{14}+l_{32}u_{24}+l_{33}u_{34}=a_{34} \qquad u_{34}=(a_{34}-l_{31}u_{14}-l_{32}u_{24})/l_{33}$$

$$⑦ l_{41}u_{14}+l_{42}u_{24}+l_{43}u_{34}+l_{44}=a_{44} \qquad l_{44}=a_{44}-l_{41}u_{14}-l_{42}u_{24}-l_{43}u_{34}$$

因此,L 和 U 矩阵的所有元素,在 l_{ii} 不等于零的情况下,可以依次由上述关系式唯一确定。现归纳为计算通式:

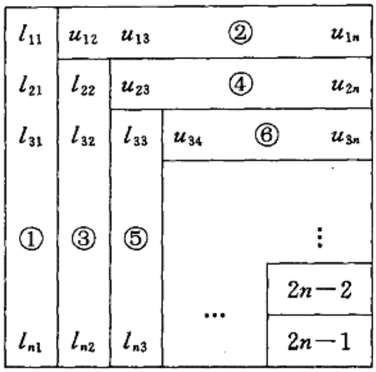
$$l_{i1} = a_{i1}, \qquad i = 1,\cdots,n \tag{1-3-8}$$

$$u_{1j} = a_{1j}/l_{11}, \qquad j = 2,\cdots,n$$

$$l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj}, \qquad \begin{cases} j = 2,\cdots,n \\ i = j,j+1,\cdots,n \end{cases} \tag{1-3-9}$$

$$u_{ij} = (a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj})/l_{ii}, \qquad \begin{cases} i = 2,\cdots,n \\ j = i+1,\cdots,n \end{cases} \tag{1-3-10}$$

分析上述过程，可以得到计算 l_{ij}和 u_{ij}的顺序如下图所示。



将上述所得 L 矩阵中各元素代入(1-3-7)式得

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

从中得出

$$\begin{cases} y_1 = b_1/l_{11} \\ y_2 = (b_2 - l_{21}y_1)/l_{22} \\ y_3 = (b_3 - l_{31}y_1 - l_{32}y_2)/l_{33} \\ y_4 = (b_4 - l_{41}y_1 - l_{42}y_2 - l_{43}y_3)/l_{44} \end{cases}$$

现归纳为计算通式

$$y_1 = b_1/l_{11} \tag{1-3-11}$$

$$y_i = (b_i - \sum_{j=1}^{i-1} l_{ij}y_j)l_{ii}, \quad i = 2,\cdots,n$$

再将所得 U 矩阵及 Y 中各元素代入(1-3-6)式得

$$\begin{bmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

从中得出

$$\begin{cases} x_4 = y_4 \\ x_3 = y_3 - u_{34}x_4 \\ x_2 = y_2 - u_{23}x_3 - u_{24}x_4 \\ x_1 = y_1 - u_{12}x_2 - u_{13}x_3 - u_{14}x_4 \end{cases}$$

现归纳为计算通式

$$x_n = y_n \tag{1-3-12}$$

$$x_i = y_i - \sum_{j=i+1}^{n} u_{ij}x_j, \quad i = n-1,\cdots,1$$

【例 1-1】 对下列矩阵 A 用 LU 法分解。

$$A=\begin{bmatrix}10 & 20 & 40 & 50\\ 2 & 5 & 10 & 14\\ 5 & 12 & 30 & 9\\ 3 & 10 & 22 & 16\end{bmatrix}$$

【解】 将 LU 法的各分解步骤分别如下

$$\textcircled{1}\begin{bmatrix}10 & & & \\ 2 & & & \\ 5 & & & \\ 3 & & & \end{bmatrix}\xrightarrow{\textcircled{2}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & & & \\ 5 & & & \\ 3 & & & \end{bmatrix}\xrightarrow{\textcircled{3}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & 1 & & \\ 5 & 2 & & \\ 3 & 4 & & \end{bmatrix}\xrightarrow{\textcircled{4}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & 1 & 2 & 4\\ 5 & 2 & & \\ 3 & 4 & & \end{bmatrix}$$

$$\xrightarrow{\textcircled{5}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & 1 & 2 & 4\\ 5 & 2 & 6 & \\ 3 & 4 & 2 & \end{bmatrix}\xrightarrow{\textcircled{6}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & 1 & 2 & 4\\ 5 & 2 & 6 & -4\\ 3 & 4 & 2 & \end{bmatrix}\xrightarrow{\textcircled{7}}\begin{bmatrix}10 & 2 & 4 & 5\\ 2 & 1 & 2 & 4\\ 5 & 2 & 6 & -4\\ 3 & 4 & 2 & -6\end{bmatrix}$$

第四节 列主元 LU 分解法及程序的编写

对于 LU 分解法，当 $a_{ii}=0$ 时，同样会使分解过程中断，为了解决这一问题和提高计算精度，可以采用选主元的方法。LU 分解选主元的方法与高斯消去法类似，通常地采用列主元法。为此需要行交换。另外，由于 $l_{i1}=a_{i1}(i=1,\cdots,n)$ 这一步计算机不需要运算，因此 LU 分解过程的顺序要作相应的变动。

对列主元 LU 分解法程序的编写，作如下几点说明。

1. 分解过程及计算公式

做第 k 步分解时

(1)首先对 k 列选主元 $|l_{k'k}|=\max\limits_{i}|l_{ik}|,i\geqslant k$，将 k' 行与 k 行交换，并将 $\boldsymbol{B}$ 中相应的行交换。

(2)然后计算

$$u_{kj}=(a_{kj}-\sum_{i=1}^{k-1}l_{ki}u_{ij})/l_{kk},\begin{cases}k=1,2,\cdots,n\\ j=k+1,k+2,\cdots,n\end{cases}$$

(3)最后计算

$$l_{ik}=a_{ik}-\sum_{j=1}^{k-1}a_{ij}u_{jk},\begin{cases}k=2,\cdots,n\\ i=k,\cdots,n\end{cases}$$

示意图

$$\begin{bmatrix}l_{11} & u_{12} & \cdots & \cdots & \cdots & \cdots & u_{1n}\\ l_{21} & l_{22} & \cdots & \cdots & \cdots & \cdots & u_{2n}\\ \vdots & \vdots & & & & & \vdots\\ l_{k1} & l_{k2} & \cdots & \cdots & l_{kk} & \cdots & a_{kn}\\ \vdots & \vdots & & \downarrow\uparrow & \text{对调} & \cdots & \vdots\\ l_{k'1} & l_{k'2} & \cdots & \cdots & l_{k'k} & \cdots & a_{k'n}\\ \vdots & \vdots & & & & & \vdots\\ l_{n1} & l_{n2} & \cdots & l_{n,k-1} & l_{nk} & \cdots & a_{nn}\end{bmatrix}\qquad\begin{bmatrix}b_1\\ b_2\\ \vdots\\ b_k\\ \uparrow\ \vdots\ \downarrow\\ b_{k'}\\ \vdots\\ b_n\end{bmatrix}$$

LU 分解：1→($k-1$)分解结果要参加行交换，$\boldsymbol{B}$ 中内容专门进行相应的行交换。

$$
\begin{bmatrix}
l_{11} & u_{12} & \cdots & \cdots & \cdots & \cdots & u_{1n} \\
l_{21} & u_{22} & \cdots & \cdots & \cdots & \cdots & u_{2n} \\
\vdots & & & & & & \vdots \\
\vdots & & l_{k'k} & & u_{k,k+1} & & u_{kn} \\
\vdots & & \vdots & & a_{k+1,k+1} & & a_{k+1,n} \\
\vdots & & \vdots & & \vdots & & \vdots \\
l_{n1} & \cdots & l_{nk} & \cdots & a_{k+1,n} & \cdots & a_{nn}
\end{bmatrix}
\quad
\begin{bmatrix}
l_{11} & u_{12} & \cdots & \cdots & \cdots & \cdots & u_{1n} \\
l_{21} & u_{22} & & & & & \\
\vdots & \vdots & & & & & \\
\vdots & \vdots & & l_{k'k} & u_{k,k+1} & \cdots & u_{kn} \\
\vdots & & \vdots & \vdots & l_{k+1,k+1} & \cdots & a_{k+1,n} \\
\vdots & \vdots & & \vdots & \vdots & & \vdots \\
l_{n1} & l_{n2} & \cdots & l_{nk} & l_{k+1,n} & \cdots & a_{nn}
\end{bmatrix}
$$

整个分解过程需要进行($n-1$)步。

此外，这里的列主元 LU 分解在进行行交换时，与列主元高斯消去法是不同的。一是对高斯消去法在做第 k 步时，只需将 $A_a^{(k-1)}$ 的剩余矩阵中第 k 行的各个列元素，与被选作主行的相应列元素进行交换，但($k-1$)列之前的列元素(以前运算结果)是不参与交换的。而在 LU 分解中则不然，因为在做第 k 步分解时，前面 1 至 k 列元素均作为 L 的元素出现，所以必须参与行交换。但这种行交换仅改变了以前计算出 1_{ij} 所在的位置，并不影响其值的大小。二是在列主元高斯消去法中，$\boldsymbol{B}$ 中内容作为增广矩阵 A_a 中第 $n+1$ 列同时参加行交换，但在列主元 LU 分解中 $\boldsymbol{B}$ 内容要专门进行与上述相应的行交换，以保证原方程不变。

$$
A_a^{(k-1)} = \left[\begin{array}{ccccc:c}
1 & a_{12}^{(1)} & & & a_{1n}^{1} & a_{1,n+1}^{(1)} \\
 & 1 & & & a_{2n}^{(2)} & a_{2,n+1}^{(2)} \\
\cdots & \cdots & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} & a_{k,n+1}^{(k-1)} \\
 & & \vdots & \downarrow\uparrow & \vdots & \downarrow\uparrow \\
 & & a_{k'k}^{(k-1)} & & a_{k'n}^{(k-1)} & a_{k',n+1}^{(k-1)} \\
 & & \vdots & & \vdots & \vdots \\
 & & a_{nk}^{(k-1)} & \cdots & a_{nn}^{(k-1)} & a_{n,n+1}^{(k-1)}
\end{array}\right]
$$

列主元高斯消去法：1→($k-1$)运算结果不参与行交换，$\boldsymbol{B}$ 中内容同时参加行交换。

2. 存贮空间

从 LU 分解过程来看，一旦 L 矩阵中第 k 列已求出，则 A 矩阵中的第 k 列元素在今后的计算中不再有用。同样，一旦 U 矩阵的第 k 行元素已经求出，则 A 矩阵的第 k 行元素在今后的计算中也不再有用。因而，我们可让 L 矩阵逐列占用 A 矩阵的下三角元素的存贮单元，同时让 U 矩阵的主对角线以上元素逐行占用 A 矩阵的主对角线以上各元素的存贮单元。整个分解过程只需在 A 矩阵所占用的存贮空间中进行。而不必占用其他存贮单元，即 A、L、U 共用存贮区示意图：

$$
\begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & & & \vdots \\
a_{n1} & \cdots & \cdots & a_{nn}
\end{bmatrix}
\xrightarrow{\text{第一步分解}}
\begin{bmatrix}
l_{11} & u_{12} & \cdots & u_{1n} \\
l_{21} & l_{22} & \cdots & a_{2n} \\
\vdots & \vdots & & \vdots \\
l_{n1} & l_{n2} & \cdots & a_{nn}
\end{bmatrix}
\cdots\cdots
$$

$$
\xrightarrow{\text{第}(n-1)\text{步分解}}
\begin{bmatrix}
l_{11} & u_{12} & \cdots & \cdots & u_{1n} \\
l_{21} & l_{22} & \cdots & \cdots & u_{2n} \\
\vdots & \vdots & & & \vdots \\
l_{n1} & l_{n2} & \cdots & \cdots & l_{nn}
\end{bmatrix}
$$