Mathematical Tripos Part IB: Easter Term 2003

Numerical Analysis – Lecture 1

1 LU factorization

1.1 Introduction

Problem 1.1 The basic problem of numerical analysis is: solve

$$Ax = b. \tag{1.1}$$

The formula

$$\det A = \textstyle\sum_i (-1)^{\sigma(i)} a_{1,i_1} a_{2,i_2} \dots a_{n,i_n}$$

gives an opportunity to solve (1.1) explicitly by Cramer's rule with the small detail that it costs $n!$ multiplications. Given a computer with 10^9 flops/sec, this way of solving an $n \times n$ system will take

a) $n = 10$, $t = 10^{-4}$ sec, b) $n = 20$, $t = 17$ min, c) $n = 30$, $t = 400\,000$ years.

Thus we have to look at methods from a constructive and practical point of view.

Example 1.2 (Triangular matrices) An *upper triangular* matrix U has $u_{ij} = 0$ if $i > j$. Then $\det U = \prod u_{ii}$. Also we can solve $Ux = y$ by so-called *back–substitution*

$$x_n = y_n/u_{nn}, \qquad x_i = (y_i - \textstyle\sum_{j=i+1}^n u_{ij}x_j)/u_{ii}, \quad i = n-1, \dots, 1.$$

Similarly, a *lower triangular* matrix L (s.t. $l_{ij} = 0$ if $i < j$) allows us to solve $Ly = b$ directly by the *forward substitution*. Hence, if we manage to factorize A as

$$A = LU,$$

then solution of the system $Ax = b$ can be split into two cases

$$Ax = LUx = b \quad \Rightarrow \quad 1)\ Ly = b, \quad 2)\ Ux = y, \tag{1.2}$$

both being solved sufficiently easily, as we have seen.

Example 1.3 (Orthogonal matrices) An orthogonal matrix Q satisfies $Q^TQ = I$, i.e., $Q^{-1} = Q^T$, so that a solution to $Qx = y$ is simply $x = Q^Ty$. If

$$A = QR$$

where Q is orthogonal and R is upper triangular, then again one can solve the system $Ax = b$ in two simple steps

$$Ax = QRx = b \quad \Rightarrow \quad 1)\ Qy = b, \quad 2)\ Rx = y.$$

1.2 Gaussian elimination

A familiar method for solving a system

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{array} \right.$$

or, in the matrix form,

$$Ax = b, \qquad A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \qquad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

is Gaussian elimination. If $a_{11} \neq 0$, then we can eliminate the first unknown x_1 from the second and subsequent equations by substracting $\frac{a_{i1}}{a_{11}} \times$ (equation1) from equation i. With

$$l_{i1} := \frac{a_{i1}}{a_{11}}, \qquad G_1 := \begin{bmatrix} 0 & & & \\ l_{21} & 0 & & \\ \vdots & & \ddots & \\ l_{n1} & & & 0 \end{bmatrix}, \qquad L_1 := I - G_1 := \begin{bmatrix} 1 & & & \\ -l_{21} & 1 & & \\ \vdots & & \ddots & \\ -l_{n1} & & & 1 \end{bmatrix},$$

this is equivalent to premultiplication of A and b by the lower triangular L_1

$$A^{(1)} = L_1 A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} \end{bmatrix}, \qquad b^{(1)} = L_1 b$$

resulting in the equivalent system $A^{(1)}x = b^{(1)}$. If $a_{22}^{(1)} \neq 0$, then we can eliminate the second unknown x_2 from the bottom $n-2$ equations, and so on. With

$$l_{ik} := \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}, \qquad G_k = \begin{bmatrix} 0 & & & & \\ & \ddots & & & \\ & & 0 & & \\ & & l_{k+1,k} & 0 & \\ & & \vdots & & \ddots & \\ & & l_{n,k} & & & 0 \end{bmatrix}, \qquad L_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -l_{k+1,k} & 1 & \\ & & \vdots & & \ddots & \\ & & -l_{n,k} & & & 1 \end{bmatrix}$$

the procedure

$$A^{(k)} = L_k A^{(k-1)} = \begin{bmatrix} a_{11} & \cdots & \cdots & \cdots & a_{1n} \\ & \ddots & & & \vdots \\ & & a_{k,k}^{(k-1)} & \cdots & a_{k,n}^{(k-1)} \\ & & 0 & a_{k+1,k+1}^{(k)} \cdots & a_{k+1,n}^{(k)} \\ & & \vdots & \vdots \quad \ddots & \vdots \\ & & 0 & a_{n,k+1}^{(k)} \cdots & a_{n,n}^{(k)} \end{bmatrix}, \qquad b^{(k)} = L_k b^{(k-1)}$$

will end with an upper triangular $A^{(n-1)} =: U$. So, with $y := b^{(n-1)}$, the resulting system

$$Ux = y$$

can be easily solved by the back substitution

$$x_n = y_n / u_{nn}, \quad x_i = (y_n - \textstyle\sum_{j=i+1}^{n} u_{ij} x_j)/u_{ii}.$$

1.3 LU factorization

It is possible to visualize the elimination process as deriving a factorization of A into two factors. Since $A^{(n-1)} = L_{n-1} \cdots L_2 L_1 A = U$, we obtain

$$A = LU, \qquad L = L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}.$$

Lemma 1.4 *We have*

$$L = L_1^{-1} \cdots L_{n-1}^{-1} = I + \sum_{k=1}^{n-1} G_k = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & \ddots & & \\ \vdots & \vdots & & 1 & \\ l_{n1} & l_{n2} & \cdots & l_{n,n-1} & 1 \end{bmatrix}.$$

Proof. Notice that $G_k G_l = 0$ if $k \le l$. This implies $(I + G_k)L_k = (I + G_k)(I - G_k) = I$, so that $L_k^{-1} = (I + G_k)$, and then, by induction say, $L = (I + G_1)(I + G_2) \cdots (I + G_{n-1}) = I + \sum_{k=1}^{n-1} G_k$. ∎

Definition 1.5 By the LU factorization of a (nonsingular) matrix A we understand a representation of A as a product

$$A = LU,$$

where L is a lower triangular matrix with unit diagonal and U is a nonsingular upper triangular matrix.

Remark 1.6 Nonsingularity of U is a natural requirement in all practical applications. Theoretically it is not necessary; see Question 1 on Examples Sheet 1.

Algorithm 1.7 The previous lemma allows computing of the LU factorization as follows.

```
for k=1 to n-1 do
    for i=k+1 to n do
        a(i,k) := a(i,k)/a(i,i);    #--    l(i,k) := a(i,k)/a(i,i);
        for m=k+1 to n do
            a(i,m) := a(i,m) - a(i,k)*a(k,m)
                                    #--    a(i,m) := a(i,m) - l(i,k)*a(k,m)
        end
    end
end
```

Notice that we can use the matrix A itself to store the elements of L and U, in the subdiagonal and in the upper triangular portions of A, respectively.

Example 1.8

$$\begin{bmatrix} 2 & -1 & -3 & 3 \\ 4 & 0 & -3 & 1 \\ 6 & 1 & -1 & 6 \\ -2 & -5 & 4 & 1 \end{bmatrix} \to \begin{bmatrix} 2 & -1 & -3 & 3 \\ 2 & 2 & 3 & -5 \\ 3 & 4 & 8 & -3 \\ -1 & -6 & 1 & 4 \end{bmatrix} \to \begin{bmatrix} 2 & -1 & -3 & 3 \\ 2 & 2 & 3 & -5 \\ 3 & 2 & 2 & 7 \\ -1 & -3 & 10 & -11 \end{bmatrix} \to \begin{bmatrix} 2 & -1 & -3 & 3 \\ 2 & 2 & 3 & -5 \\ 3 & 2 & 2 & 7 \\ -1 & -3 & 5 & -46 \end{bmatrix}$$

Application 1.9 One can use the LU factorization in the following problems.

1. Solution of linear systems (see (1.2)).
2. Calculation of the determinant: $\det A = \det L \det U = \prod u_{ii}$.
3. Calculation of the inverse of A: $A^{-1} = U^{-1} L^{-1}$.

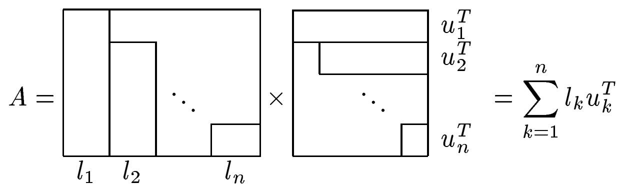
The cost. It is only multiplications and divisions that matter. By the k-th step of the LU algorithm we perform $(n-k)$ divisions and $(n-k)^2$ multiplications. Hence

$$N_{LU} = \textstyle\sum_{k=1}^{n-1} [(n-k)^2 + (n-k)] = \frac{1}{3}n^3 + \mathcal{O}(n^2).$$

In forward-substitution we use k multiplications/divisions to determine y_k, thus $N_F \sim \frac{1}{2}n^2$.

Remark 1.10 The only yet important difference between LU and Gaussian elimination is that we do not consider the right hand side b until the factorization is complete. This is useful e.g. when there are many right hand sides, in particular if not all the b's are known at the outset: in Gaussian elimination the solution for each new b would require $\mathcal{O}(n^3)$ computational operations, whereas with LU factorization $\mathcal{O}(n^3)$ operations are required for the initial factorization, but then the solution for each new b only requires $\mathcal{O}(n^2)$ operations.

Method 1.11 Another version of the LU factorization is based on the direct approach. If $A = LU$, and l_k and u_k^T are the columns and the rows of the matrices L and U respectively, then



Since the leading $k-1$ elements of l_k and u_k^T are zero for $k \geq 2$, it follows that the first $k-1$ columns and rows of the matrix $l_k u_k^T$ are zeros as well. Hence u_1^T, which is the 1-st row of the matrix $l_1 u_1^T$, coincides with the 1-st row of A, while $l_1 \cdot u_{11} = l_1 \cdot a_{11}$ coincides with the 1-st column of A. Substracting and letting $\tilde{A}^{(1)} := A - l_1 u_1^T$ we obtain similarly that u_2^T and $l_2 \cdot a_{22}^{(1)}$ are the 2-nd row and the 2-nd column of $\tilde{A}^{(1)}$ respectively, and so on.

1.4 Pivoting

The Gauss algorithms fails if $a_{kk}^{(k-1)} = 0$. A remedy is to exchange rows of $A^{(k-1)}$ by picking a suitable *pivotal equation*, i.e., the equation which is used to eliminate one unknown from certain of the other equation.

This technique is called *column pivoting* and it means that, having obtained $A^{(k-1)}$, we exchange two rows of $A^{(k-1)}$ so that the element of largest magnitude in the k-th column is in the 'pivotal position' (k,k). In other words,

$$|a_{k,k}^{(k-1)}| = \max_{j=k\ldots n} |a_{j,k}^{(k-1)}|.$$

The exchange of rows k and l can be regarded as the pre-multiplication of the relevant matrix $A^{(k-1)}$ by a permutation matrix P_{kl}. The same exchange is required in the portion of L that has been formed already (i.e., in the first $k-1$ columns). Also, we need to record the permutation of rows to solve for the right-hand side and/or to compute the determinant.

Example 1.12

$$\begin{bmatrix} 2 & 1 & 1 \\ \mathbf{4} & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix} \overset{\begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}}{\rightarrow} \begin{bmatrix} 4 & 1 & 0 \\ 2 & 1 & 1 \\ -2 & 2 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 4 & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 \\ -\frac{1}{2} & \frac{5}{2} & 1 \end{bmatrix} \overset{\begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}}{\rightarrow} \begin{bmatrix} 4 & 1 & 0 \\ -\frac{1}{2} & \frac{5}{2} & 1 \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 4 & 1 & 0 \\ -\frac{1}{2} & \frac{5}{2} & 1 \\ \frac{1}{2} & \frac{1}{5} & \frac{4}{5} \end{bmatrix},$$

$$PA = LU \quad \Rightarrow \quad A = P^T LU, \quad P = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad L = \begin{bmatrix} 1 & & \\ -\frac{1}{2} & 1 & \\ \frac{1}{2} & \frac{1}{5} & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 4 & 1 & 0 \\ & \frac{5}{2} & 1 \\ & & \frac{4}{5} \end{bmatrix}.$$

An important advantage of column pivoting is that every element of L has magnitude at most one. This avoids not just division by zero (what is not the case in the previous example) but also tends to reduce the chance of very large numbers occuring during the factorization, a phenomenon that might lead to *ill conditioning* and to accumulation of *round-off error*.

1.5 Examples

Example 1.13 (LU factorization)

$$\begin{bmatrix} -3 & 2 & 3 & -1 \\ 6 & -2 & -6 & 0 \\ -9 & 4 & 10 & 3 \\ 12 & -4 & -13 & -5 \end{bmatrix} \xrightarrow[4^{\text{th}} - 1^{\text{st}} \cdot (-4)]{\substack{2^{\text{nd}} - 1^{\text{st}} \cdot (-2) \\ 3^{\text{rd}} - 1^{\text{st}} \cdot \ \ 3}} \begin{bmatrix} -3 & 2 & 3 & -1 \\ -2 & 2 & 0 & -2 \\ 3 & -2 & 1 & 6 \\ -4 & 4 & -1 & -9 \end{bmatrix}$$

$$\xrightarrow[4^{\text{th}} - 2^{\text{nd}} \cdot \ \ 2]{3^{\text{rd}} - 2^{\text{nd}} \cdot (-1)} \begin{bmatrix} -3 & 2 & 3 & -1 \\ -2 & 2 & 0 & -2 \\ 3 & -1 & 1 & 4 \\ -4 & 2 & -1 & -5 \end{bmatrix} \xrightarrow{4^{\text{th}} - 3^{\text{rd}} \cdot (-1)} \begin{bmatrix} -3 & 2 & 3 & -1 \\ -2 & 2 & 0 & -2 \\ 3 & -1 & 1 & 4 \\ -4 & 2 & -1 & -1 \end{bmatrix}$$

i.e.

$$A = LU, \quad L = \begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ 3 & -1 & 1 & \\ -4 & 2 & -1 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} -3 & 2 & 3 & -1 \\ & 2 & 0 & -2 \\ & & 1 & 4 \\ & & & -1 \end{bmatrix}$$

Example 1.14 (Forward and back substitutions) Then the solution to the system

$$Ax = b, \qquad b = [3, -2, 2, 0]^T$$

proceeds in two steps

$$Ax = L \underbrace{Ux}_{y} = b \quad \Rightarrow \quad 1) \quad Ly = b, \quad 2) \quad Ux = y.$$

1) Forward substitution

$$\begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ 3 & -1 & 1 & \\ -4 & 2 & -1 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 2 \\ 0 \end{bmatrix} \begin{matrix} \downarrow \\ \downarrow \\ \downarrow \\ \downarrow \end{matrix} \quad \Rightarrow \quad y = \begin{bmatrix} 3 \\ 4 \\ -3 \\ 1 \end{bmatrix},$$

2) Back substitution

$$\begin{bmatrix} -3 & 2 & 3 & -1 \\ & 2 & 0 & -2 \\ & & 1 & 4 \\ & & & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ -3 \\ 1 \end{bmatrix} \begin{matrix} \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \end{matrix} \quad \Rightarrow \quad x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \end{bmatrix},$$

Example 1.15 (LU factorization with pivoting)

$$\begin{bmatrix} -3 & 2 & 3 & -1 \\ 6 & -2 & -6 & 0 \\ -9 & 4 & 10 & 3 \\ \bullet 12 & -4 & -13 & -5 \end{bmatrix} \overset{\begin{bmatrix} 4 \\ 2 \\ 3 \\ 1 \end{bmatrix}}{\rightarrow} \begin{bmatrix} 12 & -4 & -13 & -5 \\ 6 & -2 & -6 & 0 \\ -9 & 4 & 10 & 3 \\ -3 & 2 & 3 & -1 \end{bmatrix} \rightarrow \begin{bmatrix} 12 & -4 & -13 & -5 \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{5}{2} \\ -\frac{3}{4} & \bullet 1 & \frac{1}{4} & -\frac{3}{4} \\ -\frac{1}{4} & 1 & -\frac{1}{4} & -\frac{9}{4} \end{bmatrix}$$

$$\overset{\begin{bmatrix} 4 \\ 3 \\ 2 \\ 1 \end{bmatrix}}{\rightarrow} \begin{bmatrix} 12 & -4 & -13 & -5 \\ -\frac{3}{4} & 1 & \frac{1}{4} & -\frac{3}{4} \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{5}{2} \\ -\frac{1}{4} & 1 & -\frac{1}{4} & -\frac{9}{4} \end{bmatrix} \rightarrow \begin{bmatrix} 12 & -4 & -13 & -5 \\ -\frac{3}{4} & 1 & \frac{1}{4} & -\frac{3}{4} \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{5}{2} \\ -\frac{1}{4} & 1 & -\frac{1}{2} & -\frac{3}{2} \end{bmatrix} \rightarrow \begin{bmatrix} 12 & -4 & -13 & -5 \\ -\frac{3}{4} & 1 & \frac{1}{4} & -\frac{3}{4} \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{5}{2} \\ -\frac{1}{4} & 1 & -1 & 1 \end{bmatrix}$$

i.e.

$$A = P^T LU, \quad P = \begin{bmatrix} & & & 1 \\ & & 1 & \\ & 1 & & \\ 1 & & & \end{bmatrix}, \quad L = \begin{bmatrix} 1 & & & \\ -\frac{3}{4} & 1 & & \\ \frac{1}{2} & 0 & 1 & \\ -\frac{1}{4} & 1 & 1 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 12 & -4 & -13 & -5 \\ & 1 & \frac{1}{4} & -\frac{3}{4} \\ & & \frac{1}{2} & \frac{5}{2} \\ & & & 1 \end{bmatrix}$$

1.6 Exercises

Exercises with a $*$-mark are *not* subject to supervision! They are just for fun!

1.1. Calculate *all* LU factorizations of the matrix

$$A = \begin{bmatrix} 10 & 6 & -2 & 1 \\ 10 & 10 & -5 & 0 \\ -2 & 2 & -2 & 1 \\ 1 & 3 & -2 & 3 \end{bmatrix},$$

where all diagonal elements of L are one. By using one of these factorizations, find *all* solutions of the equation $Ax = b$ where $b = [-2, 0, 2, 1]^T$.

1.2. By using column pivoting if necessary to exchange rows of A, an LU factorization of a real $n \times n$ matrix A is calculated, where L has ones on its diagonal, and where the moduli of the off-diagonal elements of L do not exceed one. Let α be the largest of the moduli of the elements of A. Prove by induction on i that elements of U satisfy the condition $|u_{ij}| \leq 2^{i-1}\alpha$. Then construct 2×2 and 3×3 nonzero matrices A that yield $|u_{22}| = 2\alpha$ and $|u_{33}| = 4\alpha$ respectively.

1.3.* Let

$$A = \begin{bmatrix} 3 & 3 & 9 \\ 3 & 4 & 10 \\ 9 & 10 & 32 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 3 \\ 13 \end{bmatrix}.$$

Find the LU factorization of A and use it to solve the system $Ax = b$.

1.4.* Prove that every nonsingular matrix A admits an LU factorization with pivoting, i.e. for every nonsingular matrix A there exists a permutation matrix P such that

$$PA = LU.$$