Dense LU factorization and its error analysis

Laura Grigori

INRIA and LJLL, Sorbonne Université

February 2018

Plan

Basis of floating point arithmetic and stability analysis
- Notation, results, proofs taken from [N.J.Higham, 2002]

Direct methods of factorization
- LU factorization
  - Error analysis of LU factorization - main results
- Block LU factorization

Plan

Basis of floating point arithmetic and stability analysis

Notation, results, proofs taken from [N.J.Higham, 2002]

Direct methods of factorization

Norms and other notations

$$
\begin{aligned}
\|A\|_F &= \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2} \\
\|A\|_2 &= \sigma_{max}(A) \\
\|A\|_\infty &= \max_{1\le i\le n}\sum_{j=1}^{n}|a_{ij}| \\
\|A\|_1 &= \max_{1\le j\le n}\sum_{i=1}^{n}|a_{ij}|
\end{aligned}
$$

Inequalities $|x| \le |y|$ and $|A| \le |B|$ hold componentwise.

Floating point arithmetic

- The machine precision or unit roundoff is u
  - The maximum relative error for a given rounding procedure
  - u is of order 10^{-8} in single precision, $2^{-53} \approx 10^{-16}$ in double precision
  - Another definition: the smallest number that added to one gives a result different from one
- The evaluation involving basic arithmetic operations $+, -, *, /$ in floating point satisfies

$$fl(x\ op\ y) = (x\ op\ y)(1+\delta), \quad |\delta| \leq u$$

Relative error

- Given a real number x and its approximation $\hat{x}$, the absolute error and the relative errors are

$$E_{abs}(\hat{x}) = |x - \hat{x}|, \qquad E_{rel}(\hat{x}) = \frac{|x - \hat{x}|}{|x|} \tag{1}$$

- The relative error is scale independent
- Some examples, outline the difference with correct significant digits

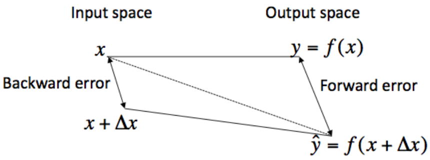
$$\begin{aligned} x &= 1.00000, \quad \hat{x} = 1.00499, \quad E_{rel}(\hat{x}) = 4.99 \times 10^{-3} \\ x &= 9.00000, \quad \hat{x} = 8.99899, \quad E_{rel}(\hat{x}) = 1.12 \times 10^{-4} \end{aligned}$$

- When x is a vector, the componentwise relative error is

$$\max_i \frac{|x_i - \hat{x}_i|}{|x_i|}$$

Backward and Forward errors

- Consider $y = f(x)$ a scalar function of a real scalar variable and $\hat{y}$ its approximation.
- Ideally we would like the forward error $E_{rel}(\hat{y}) \approx u$
- Instead we focus on the backward error, "For what set of data we have solved the problem?"
  that is we look for min $|\Delta x|$ such that $\hat{y} = f(x + \Delta x)$



Condition number

Assume f is twice continuously differentiable, then

$$\begin{aligned}\hat{y} - y &= f(x + \Delta x) - f(x) = f'(x)\Delta x + \frac{f''(x + \tau\Delta x)}{2!}(\Delta x)^2, \qquad \tau \in (0,1) \\ \frac{\hat{y} - y}{y} &= \left(\frac{xf'(x)}{f(x)}\right)\frac{\Delta x}{x} + O((\Delta x)^2)\end{aligned}$$

The condition number is

$$c(x) = \left|\frac{xf'(x)}{f(x)}\right|$$

Rule of thumb

When consistently defined, we have

$$\text{forward error} \leq \text{condition number} \times \text{backward error}$$

Preliminaries

Lemma (Lemma 3.1 in [N.J.Higham, 2002])

If $|\delta_i| \leq u$ *and* $\rho_i = \pm 1$ *for* $i = 1 : n$*, and* $nu < 1$*, then*

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = 1 + \Theta_n, \qquad |\Theta_n| \leq \frac{nu}{1-nu} = \gamma_n$$

Other notations

$$\tilde{\gamma}_n = \frac{cnu}{1-cnu}$$

Inner product in floating point arithmetic

Consider computing $s_n = x^T y$, with an evaluation from left to right. We denote different errors as $1 + \delta_i \equiv 1 \pm \delta$

$$
\begin{aligned}
\hat{s}_1 &= fl(x_1 y_1) = x_1 y_1 (1 \pm \delta) \\
\hat{s}_2 &= fl(\hat{s}_1 + x_2 y_2) = (\hat{s}_1 + x_2 y_2 (1 \pm \delta))(1 \pm \delta) \\
&= x_1 y_1 (1 \pm \delta)^2 + x_2 y_2 (1 \pm \delta)^2 \\
&\vdots \\
\hat{s}_n &= x_1 y_1 (1 \pm \delta)^n + x_2 y_2 (1 \pm \delta)^n + x_3 y_3 (1 \pm \delta)^{n-1} + \ldots + x_n y_n (1 \pm \delta)^2
\end{aligned}
$$

After applying the previous lemma, we obtain

$$
\hat{s}_n = x_1 y_1 (1 + \Theta_n) + x_2 y_2 (1 + \Theta'_n) + \ldots + x_n y_n (1 + \Theta_2)
$$

Inner product in FP arithmetic - error bounds

We obtain the following backward and forward errors

$$\begin{aligned}
\hat{s}_n &= x_1 y_1 (1+\Theta_n) + x_2 y_2 (1+\Theta'_n) + \ldots + x_n y_n (1+\Theta_2) \\
fl(x^T y) &= (x+\Delta x)^T y = x^T (y+\Delta y), |\Delta x| \leq \gamma_n |x|, |\Delta y| \leq \gamma_n |y|, \\
|x^T y - fl(x^T y)| &\leq \gamma_n \sum_{i=1}^{n} |x_i y_i| = \gamma_n |x|^T |y|
\end{aligned}$$

- High relative accuracy is obtained when computing $x^T x$
- No guarantee of high accuracy when $|x^T y| << |x|^T |y|$

Plan

Basis of floating point arithmetic and stability analysis

Direct methods of factorization
- LU factorization
- Block LU factorization

LU factorization

Compute the factorization PA = LU

Example

Given the matrix

$$A = \begin{pmatrix} 3 & 1 & 3 \\ 6 & 7 & 3 \\ 9 & 12 & 3 \end{pmatrix}$$

Let

$$M_1 = \begin{pmatrix} 1 & & \\ -2 & 1 & \\ -3 & & 1 \end{pmatrix}, \quad M_1 A = \begin{pmatrix} 3 & 1 & 3 \\ 0 & 5 & -3 \\ 0 & 9 & -6 \end{pmatrix}$$

Algebra of the LU factorization

- In general

$$
\begin{aligned}
A^{(k+1)} &= M_k A^{(k)} := \begin{pmatrix} I_{k-1} & & & & \\ & 1 & & & \\ & -m_{k+1,k} & 1 & & \\ & \dots & & \ddots & \\ & -m_{n,k} & & & 1 \end{pmatrix} A^{(k)}, \text{where} \\
M_k &= I - m_k e_k^T, \quad M_k^{-1} = I + m_k e_k^T
\end{aligned}
$$

where e_k is the k-th unit vector, $e_i^T m_k = 0, \forall i \leq k$

- The factorization can be written as

$$M_{n-1} \dots M_1 A = A^{(n)} = U$$

Algebra of the LU factorization

- We obtain

$$\begin{aligned}
A &= M_1^{-1} \dots M_{n-1}^{-1} U \\
&= (I + m_1 e_1^T) \dots (I + m_{n-1} e_{n-1}^T) U \\
&= \left(I + \sum_{i=1}^{n-1} m_i e_i^T \right) U \\
&= \begin{pmatrix} 1 & & & \\ m_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ m_{n1} & m_{n2} & \dots & 1 \end{pmatrix} U = LU
\end{aligned}$$

The need for pivoting

- For stability, avoid division by small diagonal elements
- For example

$$A = \begin{pmatrix} 0 & 3 & 3 \\ 3 & 1 & 3 \\ 6 & 2 & 3 \end{pmatrix} \tag{2}$$

  has an LU factorization if we permute the rows of matrix A

$$PA = \begin{pmatrix} 6 & 2 & 3 \\ 0 & 3 & 3 \\ 3 & 1 & 3 \end{pmatrix} = \begin{pmatrix} 1 & & \\ & 1 & \\ 0.5 & & 1 \end{pmatrix} \cdot \begin{pmatrix} 6 & 2 & 3 \\ & 3 & 3 \\ & & 1.5 \end{pmatrix} \tag{3}$$

- Partial pivoting allows to bound the multipliers $m_{ik} \leq 1$ and hence $|L| \leq 1$

Existence of the LU factorization

Theorem

Given a full rank matrix A of size $m \times n$, $m \geq n$, the matrix A can be decomposed as $A = PLU$ where P is a permutation matrix of size $m \times m$, L is a unit lower triangular matrix of size $m \times n$ and U is a nonsingular upper triangular matrix of size $n \times n$.

Proof: simpler proof for the square case. Since A is full rank, there is a permutation P_1 such that $P_1 a_{11}$ is nonzero. Write the factorization as

$$P_1 A = \begin{pmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ A_{21}/a_{11} & I \end{pmatrix} \begin{pmatrix} a_{11} & A_{12} \\ 0 & A_{22} - a_{11}^{-1} A_{21} A_{12} \end{pmatrix},$$

where $S = A_{22} - a_{11}^{-1} A_{21} A_{12}$.
Since $det(A) \neq 0$, then $det(S) \neq 0$. Continue the proof by induction on S.

Solving $Ax = b$ by using Gaussian elimination

Composed of 4 steps

1. Factor $A = PLU$, $(2/3)n^3$) flops
2. Compute $P^T b$ to solve $LUx = P^T b$
3. Forward substitution: solve $Ly = P^T * b$, n^2 flops
4. Backward substitution: solve $Ux = y$, n^2 flops

Algorithm to compute the LU factorization

- Algorithm for computing the in place LU factorization of a matrix of size $n \times n$.
- $\#flops = 2n^3/3$

for k = 1:n-1 **do**
    Let a_{ik} be the element of maximum magnitude in $A(k : n, k)$
    Permute row i and row k
    $A(k+1 : n, k) = A(k+1 : n, k)/a_{kk}$
    for $i = k+1 : n$ **do**
        for $j = k+1 : n$ **do**
            $a_{ij} = a_{ij} - a_{ik}a_{kj}$
        end for
    end for
end for

Algorithm to compute the LU factorization

- Left looking approach, pivoting ignored, A of size $m \times n$
- $\#flops = n^2 m - n^3/3$

for k = 1:n **do**
    for j = k:n **do**
        $u_{kj} = a_{kj} - \sum_{i=1}^{k-1} l_{ki} u_{ij}$
    end for
    for i = k+1:m **do**
        $l_{ik} = (a_{ik} - \sum_{j=1}^{k-1} l_{ij} u_{jk}) / u_{kk}$
    end for
end for

Error analysis of the LU factorization

Given the first $k-1$ columns of L and $k-1$ rows of U were computed, we have

$$
\begin{aligned}
a_{kj} &= l_{k1}u_{1j} + \ldots + l_{k,k-1}u_{k-1,j} + u_{kj}, j = k : n \\
a_{ik} &= l_{i1}u_{1k} + \ldots + l_{ik}u_{kk}, i = k+1 : m
\end{aligned}
$$

The computed elements of $\hat{L}$ and $\hat{U}$ satisfy:

$$
\begin{aligned}
\left| a_{kj} - \sum_{i=1}^{k-1} \hat{l}_{ki}\hat{u}_{ij} - \hat{u}_{kj} \right| &\leq \gamma_k \sum_{i=1}^{k} |\hat{l}_{ki}||\hat{u}_{ij}|, \quad j \geq k, \\
\left| a_{ik} - \sum_{j=1}^{k} \hat{l}_{ij}\hat{u}_{jk} \right| &\leq \gamma_k \sum_{j=1}^{k} |\hat{l}_{ij}||\hat{u}_{jk}|, \quad i > k.
\end{aligned}
$$

Error analysis of the LU factorization (continued)

Theorem (Theorem 9.3 in [N.J.Higham, 2002])

Let $A \in \mathbb{R}^{m\times n}, m \geq n$ and let $\hat{L} \in \mathbb{R}^{m\times n}$ and $\hat{U} \in \mathbb{R}^{n\times n}$ be its computed LU factors obtained by Gaussian elimination (suppose there was no failure during GE). Then,

$$\hat{L}\hat{U} = A + \Delta A, \qquad |\Delta A| \leq \gamma_n |\hat{L}||\hat{U}|.$$

Theorem (Theorem 9.4 in [N.J.Higham, 2002])

Let $A \in \mathbb{R}^{m\times n}, m \geq n$ and let $\hat{x}$ be the computed solution to $Ax = b$ obtained by using the computed LU factors of A obtained by Gaussian elimination. Then

$$(A + \Delta A)\hat{x} = b, \qquad |\Delta A| \leq \gamma_{3n} |\hat{L}||\hat{U}|.$$

Error analysis of $Ax = b$

Theorem (Theorem 9.4 in [N.J.Higham, 2002] continued)

$$(A + \Delta A)\hat{x} = b, \qquad |\Delta A| \leq \gamma_{3n}|\hat{L}||\hat{U}|.$$

Proof.

We have the following:

$$\begin{aligned} \hat{L}\hat{U} &= A + \Delta A, \qquad |\Delta A| \leq \gamma_n |\hat{L}||\hat{U}|, \\ (\hat{L} + \Delta L)\hat{y} &= b, \qquad |\Delta L| \leq \gamma_n |\hat{L}|, \\ (\hat{U} + \Delta U)\hat{x} &= \hat{y}, \qquad |\Delta U| \leq \gamma_n |\hat{U}|. \end{aligned}$$

Thus

$$\begin{aligned} b &= (\hat{L} + \Delta L)(\hat{U} + \Delta U)\hat{x} = (A + \Delta A_1 + \hat{L}\Delta U + \Delta L \hat{U} + \Delta L \Delta U)\hat{x} \\ &= (A + \Delta A)\hat{x}, \textit{ where} \\ |\Delta A| &\leq (3\gamma_n + \gamma_n^2)|\hat{L}||\hat{U}| \leq \gamma_{3n}|\hat{L}||\hat{U}|. \end{aligned}$$

Wilkinson's backward error stability result

Growth factor g_W defined as

$$g_W = \frac{\max_{i,j,k} |a_{ij}^k|}{\max_{i,j} |a_{ij}|}$$

Note that

$$|u_{ij}| = |a_{ij}^i| \leq g_W \max_{i,j} |a_{ij}|$$

Theorem (Wilkinson's backward error stability result, see also [N.J.Higham, 2002] for more details)

Let $A \in \mathbb{R}^{n\times n}$ and let $\hat{x}$ be the computed solution of $Ax = b$ obtained by using GEPP. Then

$$(A + \Delta A)\hat{x} = b, \qquad \|\Delta A\|_\infty \leq n^2 \gamma_{3n} g_W(n) \|A\|_\infty.$$

The growth factor

- The LU factorization is backward stable if the growth factor is small (grows linearly with n).
- For partial pivoting, the growth factor $g(n) \leq 2^{n-1}$, and this bound is attainable.
- In practice it is on the order of $n^{2/3} - n^{1/2}$

Exponential growth factor for Wilkinson matrix

$$A = diag(\pm 1)\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 1 \\ -1 & 1 & 0 & \dots & 0 & 1 \\ -1 & -1 & 1 & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & 1 \\ -1 & -1 & \cdots & -1 & 1 & 1 \\ -1 & -1 & \cdots & -1 & -1 & 1 \end{bmatrix}$$

Experimental results for special matrices

Several errror bounds for GEPP, the normwise backward error η and the componentwise backward error w ($r = b - Ax$).

$$\eta = \frac{||r||_1}{||A||_1 \; ||x||_1 + ||b||_1},$$
$$w = \max_i \frac{|r_i|}{(|A| \; |x| + |b|)_i}.$$

matrix	cond(A,2)	g_W	$\|\|L\|\|_1$	cond(U,1)	$\frac{\|\|PA-LU\|\|_F}{\|\|A\|\|_F}$	η	w_b
hadamard	1.0E+0	4.1E+3	4.1E+3	5.3E+5	0.0E+0	3.3E-16	4.6E-15
randsvd	6.7E+7	4.7E+0	9.9E+2	1.4E+10	5.6E-15	3.4E-16	2.0E-15
chebvand	3.8E+19	2.0E+2	2.2E+3	4.8E+22	5.1E-14	3.3E-17	2.6E-16
frank	1.7E+20	1.0E+0	2.0E+0	1.9E+30	2.2E-18	4.9E-27	1.2E-23
hilb	8.0E+21	1.0E+0	3.1E+3	2.2E+22	2.2E-16	5.5E-19	2.0E-17

Block formulation of the LU factorization

Partitioning of matrix A of size $n \times n$

$$A = \left[\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \end{array} \right]$$

where A_{11} is of size $b \times b$, A_{21} is of size $(m-b) \times b$, A_{12} is of size $b \times (n-b)$ and A_{22} is of size $(m-b) \times (n-b)$.

Block LU algebra

The first iteration computes the factorization:

$$P_1^T A = \left[\begin{array}{cc} L_{11} & \\ L_{21} & I_{n-b} \end{array} \right] \cdot \left[\begin{array}{cc} I_b & \\ & A^1 \end{array} \right] \cdot \left[\begin{array}{cc} U_{11} & U_{12} \\ & I_{n-b} \end{array} \right]$$

The algorithm continues recursively on the trailing matrix A^1.

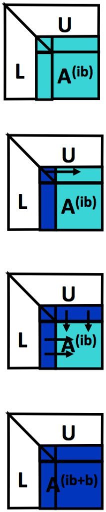
Block LU factorization - the algorithm

1. Compute the LU factorization with partial pivoting of the first block column
$$P_1 \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = \begin{pmatrix} L_{11} \\ L_{21} \end{pmatrix} U_{11}$$
2. Pivot by applying the permutation matrix P_1^T on the entire matrix,
$$\bar{A} = P_1^T A.$$
3. Solve the triangular system
$$L_{11} U_{12} = \bar{A}_{12}$$
4. Update the trailing matrix,
$$A^1 = \bar{A}_{22} - L_{21} U_{12}$$
5. Compute recursively the block LU factorization of A^1.

LU Factorization as in ScaLAPACK

LU factorization on a P = Pr x Pc grid of processors
For ib = 1 to n-1 step b
$A(ib) = A(ib : n, ib : n)$

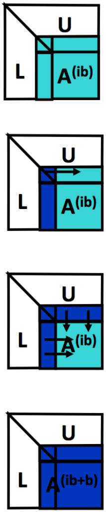
1. Compute panel factorization
   - find pivot in each column, swap rows
2. Apply all row permutations
   - broadcast pivot information along the rows
   - swap rows at left and right
3. Compute block row of U
   - broadcast right diagonal block of L of current panel
4. Update trailing matrix
   - broadcast right block column of L
   - broadcast down block row of U



Cost of LU Factorization in ScaLAPACK

LU factorization on a P = Pr x Pc grid of processors
For ib = 1 to n-1 step b
$A(ib) = A(ib : n, ib : n)$

1. Compute panel factorization
   - $\#messages = O(n \log_2 P_r)$
2. Apply all row permutations
   - $\#messages = O(n/b(\log_2 P_r + \log_2 P_c))$
3. Compute block row of U
   - $\#messages = O(n/b \log_2 P_c)$
4. Update trailing matrix
   - $\#messages = O(n/b(\log_2 P_r + \log_2 P_c)$



Cost of parallel block LU

Consider that we have a $\sqrt{P} \times \sqrt{P}$ grid, block size b

$$\gamma \cdot \left(\frac{2/3 n^3}{P} + \frac{n^2 b}{\sqrt{P}} \right) + \beta \cdot \frac{n^2 \log P}{\sqrt{P}} + \alpha \cdot \left(1.5 n \log P + \frac{3.5 n}{b} \log P \right).$$

Acknowledgement

- Stability analysis results presented from [N.J.Higham, 2002]
- Some of the examples taken from [Golub and Van Loan, 1996]

References (1)

Golub, G. H. and Van Loan, C. F. (1996).
Matrix Computations (3rd Ed.).
Johns Hopkins University Press, Baltimore, MD, USA.

N.J.Higham (2002).
Accuracy and Stability of Numerical Algorithms.
SIAM, second edition.

Schreiber, R. and Loan, C. V. (1989).
A storage efficient WY representation for products of Householder transformations.
SIAM J. Sci. Stat. Comput., 10(1):53–57.