Homework 2 Solutions

by WeiHao Li

Fudan University

(Numerical Algorithms with Case Studies I)

Problem 1. Denote the matrix mentioned as A. To eliminate the first column, let

$$L_1 = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & & 1 \end{pmatrix},$$

then

$$U_1 = L_1^{-1} A = \begin{pmatrix} 1 & 0 & 0 & \cdots & 1 \\ 0 & 1 & 0 & \cdots & 2 \\ 0 & 1 & 1 & \cdots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 1 & & 2 \end{pmatrix}.$$

Suppose we have eliminate the first k column of A:

$$U_k = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 2 \\ 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 4 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 2^k \\ 0 & 0 & 0 & \cdots & 1 & 1 & \cdots & 2^k \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 1 & \cdots & 2^k \end{pmatrix},$$

let

$$L_{k+1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \end{pmatrix},$$

then

$$U_{k+1} = L_{k+1}^{-1} U_k = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 2 \\ 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 4 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 2^k \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 2^{k+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 2^{k+1} \end{pmatrix}.$$

Repeat this procedure for $n-1$ times, let

$$L=L_1L_2\cdots L_{n-1},\quad U=U_{n-1}$$

we conclude that

$$A=LU=\begin{pmatrix}1&0&0&\cdots&0\\1&1&0&\cdots&0\\1&1&1&\cdots&0\\\vdots&\vdots&\vdots&\ddots&\vdots\\1&1&1&\cdots&1\end{pmatrix}\begin{pmatrix}1&0&0&\cdots&1\\0&1&0&\cdots&2\\0&0&1&\cdots&4\\\vdots&\vdots&\vdots&\ddots&\vdots\\0&0&0&\cdots&2^{n-1}\end{pmatrix}$$

Problem 2. Let $A=\{a_{ij}\}$, $i,j=1\cdots n$. For arbitary $i,j>1$, after one step of Gaussian elimination:

$$a_{ij}\rightarrow a_{ij}-\frac{a_{1j}a_{i1}}{a_{11}},\quad a_{ji}\rightarrow a_{ji}-\frac{a_{1i}a_{j1}}{a_{11}}.$$

Because $a_{ij}=\overline{a_{ji}}$ for any i,j, then

$$a_{ji}-\frac{a_{1i}a_{j1}}{a_{11}}=\overline{a_{ij}}-\frac{\overline{a_{1j}a_{i1}}}{a_{11}}=\overline{a_{ij}-\frac{a_{1j}a_{i1}}{a_{11}}}$$

so the (n − 1) × (n − 1) trailing principal submatrix is still Hermitian.

Problem 3. Let $A=\{a_{ij}\}$, $i,j=1\cdots n$. For arbitary $i,j>1$, after one step of Gaussian elimination:

$$a_{ij}\rightarrow a_{ij}-\frac{a_{1j}a_{i1}}{a_{11}},\quad a_{ii}\rightarrow a_{ii}-\frac{a_{1i}a_{i1}}{a_{11}}(i\neq j)$$

then

$$\left|a_{ij}-\frac{a_{1j}a_{i1}}{a_{11}}\right|\leqslant|a_{ij}|+\frac{|a_{1j}||a_{i1}|}{|a_{11}|},$$

$$\left|a_{ii}-\frac{a_{1i}a_{i1}}{a_{11}}\right|\geqslant|a_{ii}|-\frac{|a_{1i}||a_{i1}|}{|a_{11}|},$$

and for A is strictly diagonally dominant,

$$\begin{aligned}&|a_{ii}|-\frac{|a_{1i}||a_{i1}|}{|a_{11}|}-\sum_{2\leqslant j\leqslant n,j\neq i}\left(|a_{ij}|+\frac{|a_{1j}||a_{i1}|}{|a_{11}|}\right)\\=\;&|a_{ii}|-\sum_{2\leqslant j\leqslant n,j\neq i}|a_{ij}|-\frac{(\sum_{2\leqslant k\leqslant n}|a_{1k}|)|a_{i1}|}{|a_{11}|}\\>\;&|a_{ii}|-\sum_{2\leqslant j\leqslant n,j\neq i}|a_{ij}|-|a_{i1}|\\=\;&|a_{ii}|-\sum_{1\leqslant j\leqslant n,j\neq i}|a_{ij}|>0.\end{aligned}$$

so we conclude that the (n − 1) × (n − 1) trailing principal submatrix is still strictly diagonally dominant.

Problem 4. L is an lower banded matrix with a lower bandwidth $b+1$, and U is an upper banded matrix with an upper bandwidth $b+1$. That means, the places which are zeros in A will also be zeros in L and U.

To prove this, see when eliminate the first column, we just need to add the first row to the 2, $3, \cdots (b+1)$th row, which means non-zero numbers will only appear on the $2, 3, \cdots (b+1)$th row in the first column of L_1; and since only the $1, 2, \cdots (b+1)$th of the first row of A are non-zero, the elimination of first column won't ruin zeros in rows below in the upper triangular part of U_1. After eliminating the first column, the size $n-1$ trailing principal submatrix is still a banded matrix with bandwidth $2b+1$, so statements above hold for further elimination.Then we can see the final L and U are also banded.

For the first $n-b$ column, the complexity of computing the elimination of one column is equivalent to eliminating the first column of a $b+1$ size full square matrix; for the last b column, the total complexity is equivalent to elimination of a b size full square matrix. so the total complexity is

$$
\begin{aligned}
& 2b(b+1)\,(n-b) + \sum_{k=1}^{b-1} \left((b-k) + 2(b-k)^2\right) \\
= \; & (2b^2+2b)n - \frac{4}{3}b^3 - \frac{5}{2}b^2 - \frac{1}{6}b \\
= \; & (2b^2+2b)n + O(b^3)
\end{aligned}
$$

Problem 5. See solinear.py for implementation and visualization. The log-log plot visualize the execution time of program in terms of matrix size is shown below.