CSC721 Algorithms Fall 2017

LUP Decomposition

Let A be a non-singular $n \times n$ matrix where $n = 2^k$ for some $k \geq 0$. The LUP-decomposition of A is computed by a recursive divide-and-conquer algorithm. The top-level call to start the computation is:

```
FACTOR( A, n, n )
```

As we descend the levels of recursion, the algorithm works by dividing the rows of A into submatrices. For this reason, we assume that the input to FACTOR is a $m \times p$ (rectangular) matrix with fewer rows than columns, i.e., $m \leq p$.

$(L, U, P) =$ FACTOR(A, m, p)
{
  if ($m == 1$) {
    Let L be a 1×1 matrix, $L = [1]$;
    Find a column c of A such that $A_{1,c} \neq 0$;
    Let P be the $p \times p$ permutation matrix that exchanges column 1 and c. Note $P = P^{-1}$.
    Let $U = AP$;
    return (L, U, P) ;
  }
  else {
    Partition A into two blocks B and C, both size $m/2 \times p$; /* See figure (a). */
    $(L_1, U_1, P_1) =$ FACTOR($B, m/2, p$) ;
    $D = CP_1^{-1}$; /* See figure (b). */
    Let E be the first $m/2$ columns of U_1 ;
    Let F be the first $m/2$ columns of D ; /* See figure (c). */
    $G = D - FE^{-1}U_1$; /* See figure (d). */
    Let G' be the rightmost $p - m/2$ columns of G ;
    $(L_2, U_2, P_2) =$ FACTOR($G', m/2, p - m/2$) ;
    Let P_3 be the $p \times p$ permutation matrix with $I_{m/2 \times m/2}$ in the upper left and
      P_2 in the lower right ; /* See figure (e). */
    $H = U_1 P_3^{-1}$; /* See figure (f). */
    Let L be the $m \times m$ matrix consisting of L_1, $0_{m/2 \times m/2}$, FE^{-1}, and L_2 ; /* See figure (g). */
    Let U be the $m \times p$ matrix with H in the upper block and $0_{m/2 \times m/2}$ joined to U_2
      in the lower block ; /* See figure (g). */
    $P = P_3 P_1$; /* See figure (g). */
    return (L, U, P) ;
  }
}

To analyze the LUP algorithm, we assume that the matrix size n is a power of two ; i.e., $n = 2^\ell$ for some $\ell \geq 0$. The formal parameter m is initially n and is always divided by two at each level in the recursion. Therefore, $m = 2^k$ for some $k \geq 0$. Let $T(m)$ denote the time taken by algorithm FACTOR() and let $M(m)$ denote the time needed to multiply two $m \times m$ matrices. In the proof of an upper bound for $T(m)$, we need the following hypothesis:

$$\text{There exists a small positive number } \epsilon > 0 \text{ such that:} \quad 2^{2+\epsilon} M(m) \leq M(2m) \tag{1}$$

Roughly speaking, equation (1) says that $M(m)$ grows slightly faster than Ωm^2. By carefully counting the time taken by each step in the pseudo-code for FACTOR, we arrive at the following recurrence inequality:

$$\begin{array}{rcl} T(1) & \leq & bn \qquad // \text{ Since } p \leq n. \\ T(m) & \leq & 2T(m/2) + \frac{en}{m} M(m/2) \end{array} \tag{2}$$

where e is a constant.

We solve the recurrence inequality by writing out the usual table of inequalities.

$$\begin{array}{rcl} T(m) & \leq & 2T(m/2) \,+\, en/m\, M(m/2) \\ T(m/2) & \leq & 2T(m/4) \,+\, 2en/m\, M(m/4) \\ T(m/4) & \leq & 2T(m/8) \,+\, 2^2 en/m\, M(m/8) \\ . & & \\ . & & \\ . & & \\ T(m/2^{k-1}) & \leq & 2T(1) \,+\, 2^{k-1} en/m\, M(m/2^k) \end{array}$$

We need to multiply each equation above by the appropriate power of two so that cancellation will occur when we sum.

$$\begin{array}{lrcl} & T(m) & \leq & 2T(m/2) \,+\, en/m\, M(m/2) \\ + & 2\,T(m/2) & \leq & 2^2 T(m/4) \,+\, 2en/m\, M(m/4) \cdot 2 \\ + & 2^2\,T(m/4) & \leq & 2^3 T(m/8) \,+\, 2^2 en/m\, M(m/8) \cdot 2^2 \\ + & . & & \\ + & . & & \\ + & . & & \\ + & 2^{k-1}\,T(\frac{m}{2^{k-1}}) & \leq & 2^k T(1) \,+\, 2^{k-1} en/m\, M(m/2^k) \cdot 2^{k-1} \\ & \text{---------} & & \\ & T(m) & \leq & bmn \,+\, \frac{en}{m} \left[M(m/2) + 4M(m/4) + 4^2 M(m/8) + ... + 4^{k-1}\, M(m/2^k) \right] \\ & & = & bmn \,+\, \frac{en}{4m} \left[4M(m/2) + 4^2 M(m/4) + 4^3 M(m/8) + ... + 4^k M(m/2^k) \right] \end{array} \tag{3}$$

The sum in the last expression above can be simplified using the first part of our hypothesis regarding $M(m)$. Recall:

$$2^{2+\epsilon}\, M(m) \,\leq\, M(2m) \tag{4}$$

We can re-formulate inequality (4) as:

$$M(m/2) \,\leq\, \frac{1}{2^{2+\epsilon}}\, M(m) \,=\, \frac{1}{4} \cdot \frac{1}{2^\epsilon}\, M(m) \tag{5}$$

We can also generalize inequality (5) to j factors of 2 in the denominator.

$$M(m/2^j) \leq \frac{1}{4^j} \cdot \frac{1}{(2^\epsilon)^j} M(m) \tag{6}$$

We apply inequality (6) to each term in the last result from equation (3) to obtain:

$$T(m) \quad \leq \quad bmn + \tfrac{en}{4m} M(m) \left[\tfrac{1}{2^\epsilon} + \tfrac{1}{(2^\epsilon)^2} + \tfrac{1}{(2^\epsilon)^3} + \ldots + \tfrac{1}{(2^\epsilon)^k} \right] \tag{7}$$

We observe that the summation in equation (7) is a convergent series and is bounded by a constant. We combine that constant with e, and the factor of 4 in the denominator into a new constant, $\hat{e}$ to obtain:

$$\begin{aligned} T(m) &\leq bmn + \tfrac{\hat{e}n}{m} M(m) \\ &\in \mathcal{O}\left(\tfrac{n}{m} M(m) \right) \end{aligned} \tag{8}$$

At the top level call, we have $m = n$. Therefore, $T(n) \in \mathcal{O}(M(n))$.