Example of Matrix Multiplication by Fox Method

Thomas Anastasio

November 23, 2003

Fox's algorithm for matrix multiplication is described in Pacheco[1]. This handout gives an example of the algorithm applied to 2×2 matrices, A and B. The product is a 2×2 matrix C.

$$A = \begin{vmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{vmatrix} \qquad B = \begin{vmatrix} B_{00} & B_{01} \\ B_{10} & B_{11} \end{vmatrix} \qquad C = \begin{vmatrix} A_{00}B_{00} + A_{01}B_{10} & A_{00}B_{01} + A_{01}B_{11} \\ A_{10}B_{00} + A_{11}B_{10} & A_{10}B_{01} + A_{11}B_{11} \end{vmatrix}$$

Assume that we have n^2 processes, one for each of the elements in A, B, and C. Call the processes P_{00}, P_{01}, P_{10}, and P_{11}, and think of them as being arranged in a grid as follows:

P_{00}	P_{01}
P_{10}	P_{11}

Allocate space on each processor P_{ij} for an A element, a B element, and a C element.

Fox's algorithm takes n stages for matrices of order n. The algorithm starts off with each $C_{i,j} = 0$. In stage k, process $P_{i,j}$ computes

$$C_{i,j} = C_{i,j} + A_{i,i+k} \times B_{i+k,j}$$

In this example, since our matrices are of order 2, there will be two stages. In stage 0, $P_{i,j}$ computes $C_{i,j} = C_{i,j} + A_{i,i} \times B_{i,j}$. In stage 1, $P_{i,j}$ computes $C_{i,j} = C_{i,j} + A_{i,i+1} \times B_{i+1,j}$, a column to the "right" in A and a row "down" in B.

1. Stage 0

   (a) We want $A_{i,i}$ on process $P_{i,j}$, so broadcast the diagonal elements of A across the rows, $(A_{ii} \rightarrow P_{ij})$. This will place $A_{0,0}$ on each $P_{0,j}$ and $A_{1,1}$ on each $P_{1,j}$. The A elements on the P matrix will be

A_{00}	A_{00}
A_{11}	A_{11}

   (b) We want $B_{i,j}$ on process $P_{i,j}$, so broadcast B across the rows $(B_{ij} \rightarrow P_{ij})$. The A and B values on the P matrix will be

A_{00} B_{00}	A_{00} B_{01}
A_{11} B_{10}	A_{11} B_{11}

[1] Peter Pacheco, *Parallel Programming with MPI*, Morgan-Kaufmann, 1996, Section 7.2

(c) Compute $C_{ij} = AB$ for each process

$$\begin{array}{c|c} A_{00} & A_{00} \\ B_{00} & B_{01} \\ C_{00} = A_{00}B_{00} & C_{01} = A_{00}B_{01} \\ \hline A_{11} & A_{11} \\ B_{10} & B_{11} \\ C_{10} = A_{11}B_{10} & C_{11} = A_{11}B_{11} \end{array}$$

We are now ready for the second stage. In this stage, we broadcast the next column (mod n) of A across the processes and shift-up (mod n) the B values.

2. Stage 1

(a) The next column of A is $A_{0,1}$ for the first row and $A_{1,0}$ for the second row (it wrapped around, mod n). Broadcast next A across the rows

$$\begin{array}{c|c} A_{01} & A_{01} \\ B_{00} & B_{01} \\ C_{00} = A_{00}B_{00} & C_{01} = A_{00}B_{01} \\ \hline A_{10} & A_{10} \\ B_{10} & B_{11} \\ C_{10} = A_{11}B_{10} & C_{11} = A_{11}B_{11} \end{array}$$

(b) Shift the B values up. $B_{1,0}$ moves up from process $P_{1,0}$ to process $P_{0,0}$ and $B_{0,0}$ moves up (mod n) from $P_{0,0}$ to $P_{1,0}$. Similarly for $B_{1,1}$ and $B_{0,1}$.

$$\begin{array}{c|c} A_{01} & A_{01} \\ B_{10} & B_{11} \\ C_{00} = A_{00}B_{00} & C_{01} = A_{00}B_{01} \\ \hline A_{10} & A_{10} \\ B_{00} & B_{01} \\ C_{10} = A_{11}B_{10} & C_{11} = A_{11}B_{11} \end{array}$$

(c) Compute $C_{ij} = AB$ for each process

$$\begin{array}{c|c} A_{01} & A_{01} \\ B_{10} & B_{11} \\ C_{00} = C_{00} + A_{01}B_{10} & C_{01} = C_{01} + A_{01}B_{11} \\ \hline A_{10} & A_{10} \\ B_{00} & B_{01} \\ C_{10} = C_{10} + A_{10}B_{00} & C_{11} = C_{11} + A_{10}B_{01} \end{array}$$

The algorithm is complete after n stages and process $P_{i,j}$ contains the final result for $C_{i,j}$.