Matrix Multiplication

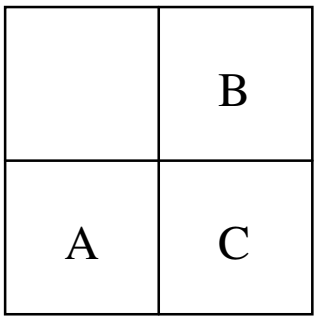
Definition: Given $A_{n\times n}$ and $B_{n\times n}$, determine $C_{n\times n} = A_{n\times n} \times B_{n\times n}$, where $c_{i,j} = \sum_{k=1}^{n} a_{i,k} b_{k,j}$.

RAM: A traditional sequential algorithm will multiply $A_{n\times n} \times B_{n\times n} \Rightarrow C_{n\times n}$ in $\Theta(n^3)$ time. {Note that better sequential algorithms do exist.}

PRAM: Discuss tradeoffs in no. of processors, size of memory, and running time.

Mesh: Discuss implementations on $n \times n$ mesh: Suppose we are given a $2n \times 2n$ mesh such that *A* is stored in the lower-left quadrant, and *B* is stored in the upper-right quadrant, and *C* is to be produced in the lower-right quadrant, matrix multiply *A* and *B* to get *C*.

Do:



for $2n \times 2n$ mesh

a) Initialize $c_{i,j} \leftarrow 0 \ \ \forall i, j$

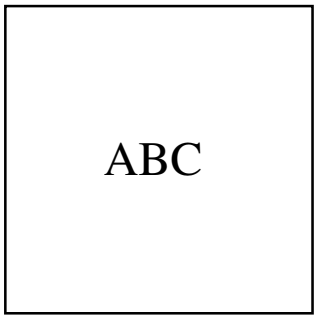
b) For *k*=1 to *n* do
   i) In lock-step, row *k* of *A* begins moving left
   ii) In lock-step, column *k* of *B* begins moving down
   iii) When the PE storing $c_{i,j}$ receives an *a* from the left and a *b* from the top, it performs $c_{i,j} \leftarrow c_{i,j} + ab$. In the next step, it continues passing $a \rightarrow$ and $b \downarrow$.

c) For *k*=*n*+1 to 3*n*-2 do step 2c.

Copyright © 1996-1997 by Russ Miller. All Rights Reserved.

Suppose we are given an $n \times n$ mesh with $a_{i,j}, b_{i,j}, c_{i,j}$ all stored in processor $P_{i,j}$. Matrix multiply *A* and *B* to get *C*.

Discuss in **Detail:**



for $n \times n$ mesh

Show the proper timings of initializing the row/column rotations to do the simulation of the systolic (external input) version.

1. Can flip *B* over and *A* around so that we can simulate above with row and column rotations, or
2. Don't flip, but follow above conceptually to get same result.

Copyright © 1996-1997 by Russ Miller. All Rights Reserved.