Parallel Algorithms and Parallel Computers

(in4026)

Examination June 28, 2010

14.00-17.00 hrs

This exam consists of two parts: Part I (architecture) and Part II (algorithms). If you have done Part I (April 2010), you can do only Part II or both Part I and Part II. If you have not done Part I in April 2010, you have to do both parts.

Please do note that after the August 2010 re-examination, the individual results for Part I and Part II obtained in 2010 or earlier are no longer valid!

> *IMPORTANT: Please use a separate sheet for each question!*

PART I

Question 1 *(25 points)*

a. Given is a multistage interconnection network with p processors. Every switching element in the network consists of a 8 by 8 crossbar switch. The interconnection network has 2 stages. How many processors can be connected to the multistage network to make an arbitrary connection between the processors possible?

b. Give three arguments for selecting a network with a mesh topology.

c. Given is a 2D mesh without wrap-around connections met p processors. Give expressions for the diameter, the bisection width, the arc connectivity, and the number of links.

d. Given is a ring with p processors. What is the maximal communication time of a message of length m if we use *Cut-Through* routing.

e. Define the ideal model of a parallel computer. Give four subclasses of this model with their properties. Which subclass is the most powerful one and why?

Question 2 *(25 points)*

a. Show how to embed a p-node 3D-dimensional Mesh into a p-node Hypercube. What are the allowable values of the mesh dimensions for your embedding

b. Given is a 2D mesh without wrap around connections. Suppose $P_{Sx,Sy}$ denotes the source processor and $P_{Dx,Dy}$ the destination processor.

- Explain the working of the XY routing algorithm in a 2D mesh.
- What is the length of the path as function of *Sx, Sy, Dx* and *Dy*?
- Is this routing algorithm minimal?

c. Given is a p processor ring. What is the communication time of a message of length m for a *one-to-all* broadcast using *store-and-forward* routing.

d. Given is a problem with workload W that has a sequential component W_s. The parallel component W_p is totally parallelizable. Proof that W/W_s is an upperbound to the speedup S.

e. Given is that the cost of a parallel algorithm and its workload are $C = O(n \log n)$ and $W = \log n$, respectively. Is this parallel algorithm Cost Optimal? Explain your answer.

Part II

Question 3 *(20 points)*

Let x be an integer and let $A[1..n]$ be an arbitrary array of integers.

a. *(5 points)* Give a fast parallel algorithm on a CRCW to check whether x occurs as one of the integers in A. What is the work-time complexity of your algorithm?

b. *(5 points)* Give a $T(n) = O(\log n)$ and $W(n) = O(n)$ algorithm on a CREW that, given x and A, computes $rank[x:A]$.
Notice that it is **not** required that A is sorted.

c. *(10 points)* Suppose that it is known that A is a sorted array.
Provide a parallel algorithm on a CREW that computes $rank[x:A]$ in $o(\log n)$ time and $O(n)$ work.

Question 4 *(10 points)*

Consider the following (third-order) linear recurrence

$$y_0 = 1$$
$$y_1 = 2$$
$$y_2 = 2$$
$$y_i = y_{i-1} + 3y_{i-2} - 2y_{i-3} \text{ for } i = 3, \ldots, n-1.$$

Let $\mathbf{y} = [y_0, \ldots, y_{n-1}]$.

a. *(5 points)* Show how this linear recurrence can be reduced to a first-order linear recurrence.

b. *(5 points)* Show that there exists a parallel algorithm to compute $\mathbf{y}$ in time $T(n) = O(\log n)$. What is the work-complexity of your algorithm?

Question 5 *(20 points)*

a. *(5 points)* Give the time-work complexity of the doubly-logarithmic tree algorithm for computing a maximum of an array A of n integers. Why is this algorithm not weakly optimal?

b. *(5 points)* There is an algorithm that is able to reduce the problem of finding the maximum in an array by reducing it to finding the maximum of an array of $n/(\log\log n)$ numbers without affecting the solution. The time complexity of this algorithm is $O(\log \log n)$ and its work complexity $O(n)$.
Explain how such an algorithm could be obtained by partitioning the array A.
You only have to explain the idea of the algorithm.

c. *(10 points)* Give a combination of this algorithm and the doubly-logarithmic tree algorithm to obtain an algorithm with $O(\log \log n)$ time complexity and $O(n)$ work complexity and justify its complexity.

End of Examination