Université Grenoble Alpes, Grenoble INP, UFR IM^{2}AG

Master 1 Informatique and Master 1 MOSIG

UE Parallel Algorithms and Programming
TD # 4

F. Desprez (Inria), J-F. Méhaut (UGA/Polytech), T. Ropars (UGA/IM^{2}AG), E. Saillard (Inria)

April 25, 2017

Exercise 1: Pipelining

We consider a single row of n processes. We note M the size of the message; L the latency on a link; B the bandwidth of a link. We want to understand how pipelining helps improving the time to transfer the message from the first process to the last process in the row. We assume store-and-forward communication.

1. Compute the time required to send the message as one single block to the last process in the row.
2. To optimize performance, we apply pipelining by dividing the message into r packets. Compute the new transfer time. To better understand how this may work, we suggest you to try reasoning on an example with, for instance, $n = 4$ and $r = 5$.

Exercise 2: Network topology

We consider the network topology described in Figure 1.

1. How many nodes and links are included in the topology?
2. What is the degree of the nodes and the diameter of the network?
3. What is the bisection width of this network? We recall that the bisection width can be defined as the minimum number of links to cut to partition the network into two parts of equal size.

Exercise 3: Matrix transposition on a ring of processors

Transposing matrix A is exchanging element A_{ij} with element A_{ji} for all i and j.

In this exercice, we want to develop a parallel algorithm to transpose a $n \times n$ matrix. We suppose the matrix data are already distributed over the processors: A block of rows is associated with each processor.

On a ring of p processors, we assume 2-way links between processors. We also assume that p divides n.

1. For a ring of size 4, describe the message exchanges required for transposing the matrix. What kind of global communication is it?
2. To derive an algorithm for matrix transposition, give the messages exchanges required on rings of size 3, 4, 5 and 6. Group the messages in phases and compute the execution time of your algorithm.

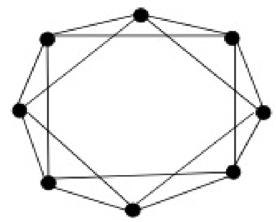

Figure 1: Network topology

Exercise 4: Matrix transposition on a grid of processors

We consider a $p = q \times q$ grid of processors. The matrix data are already distributed over the grid: Each processor has a sub-matrix of size $\frac{n}{q} \times \frac{n}{q}$ (we assume q divides n).

We assume processors can communicate in the four directions at the same time.

1. Considering a 6×6 grid, describe the communication required to transpose the matrix.
2. Derive the complexity of the general algorithm.