**ITS 470**

**Homework 4**

**Geyang Xie**

Please put Course number, your Full Name, Assignment number on top right. Submit your homework on Blackboard. Please name your file as “ITS470HW4FirstNameLastName.(doc or docx)”. Any failure to do so will result in deduction.

1. \*Consider the all-reduce operation in which each processor starts with an array of *m* words, and needs to get the global sum of the respective words in the array at each processor. This operation can be implemented on a ring using one of the followings. Note that log function in this chapter has base of 2, not 10. (i.e., log p is actually log2 p)
   1. All-to-all broadcast of all the arrays followed by a local computation of the sum of the respective elements of the array
   2. Single node accumulation of the elements of the array, followed by a one-to-all broadcast of the result array.
2. For each of the above cases, compute the run time in terms of *m*, *ts*, and *tw*. (5 points)

T = (ts + twm)(p-1)

It should do all-to-one reduction and do one-to-all broadcast

T = 2x(ts + twm)log2(p)

1. Assume that *ts* = 100, *tw* = 1, and m is very large (i.e., 100). Which one is better? Justify your answer by comparing them in the plot with the number of processors (i.e., 2, 4, 6, …, and 20). (10 points)

P = 2, all to all = 200, one to all = 400

P = 4, all to all = 600, one to all = 800

P = 6, all to all = 1000, one to all = 1000

P = 8, all to all = 1400, one to all = 1200

P = 10, all to all = 1800, one to all = 1300

P = 12, all to all = 2200, one to all = 1400

P = 14, all to all = 2600, one to all = 1500

P = 16, all to all = 3000, one to all = 1600

P = 18, all to all = 3400, one to all = 1670

P = 20, all to all = 3800, one to all = 1728

Assume that *ts* = 100, *tw* = 1, and m is very small (i.e., 1). Which one is better? Justify your answer by comparing them in the plot with the number of processors (i.e., 2, 4, 6, …, and 20). (10 points)

P = 2, all to all = 101, one to all = 202

P = 4, all to all = 303, one to all = 404

P = 6, all to all = 505, one to all = 522

P = 8, all to all = 707, one to all = 606

P = 10, all to all = 909, one to all = 671

P = 12, all to all = 1111, one to all = 724

P = 14, all to all = 1313, one to all = 769

P = 16, all to all = 1515, one to all = 808

P = 18, all to all = 1717, one to all = 842

P = 20, all to all = 1919, one to all = 873

1. Explain how Scatter is different from one-to-all broadcast. (5 points)

Scatter- one node sends out a message that i the size of m to every other node

one-to-all broadcast-one processor has a piece of data size m to send to everyone.

1. \*All-to-all personalized communication can be optimally done in hypercube. Considering 8-node hypercube, find the partners of node 3 at step 4, node 7 at step 6, and node 5 at step 7. (5 points)

Node 3 at step 4: 011 XOR 100 = 111 node 7

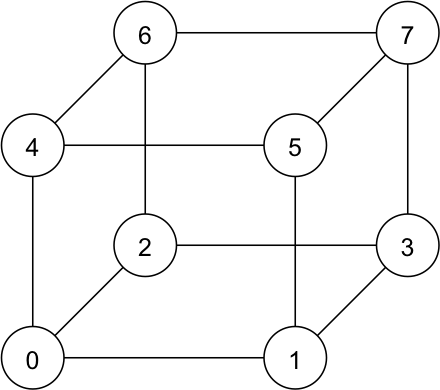
Node 7 at step 6: 111 XOR 110 = 001 node 1

Node 5 at step 7: 101 XOR 111 = 010 node 2

1. In 64-node spare wraparound mesh, each processor needs to send a message to one particular processor in the distance of 50 nodes. Explain how it can be done in step by step. (5 points)

The first step is to rotate the values in the rows to the right. The second step is to rotate the 1 st column all the values up one position.. The third step is to column shift all of the columns up once. The fourth step is the final distribution of the data.

1. \*In 8-node hypercube as shown below, each processor needs to send a message to one particular processor, which is in the distance of 5 nodes. Explain how the message from processor 1 would route to processor 6. (10 points)



node 1 (001) to node 6 (110).

001 XOR 110 = 111, to x direction, then it goes to node 0 (000).

000 XOR 110 = 110, to y direction, it goes to node 2 (010).

010 XOR 110 = 100, to z direction, it goes to node 6 (110).

110 XOR 110 = 000, arrived.

1. \*Assume that the serial runtime is approximately 15 minutes and average parallel runtime is approximately 5 minutes. If there are 5 processors are used in parallel computing, find followings. Assume that the best sequential program is used as a baseline. (15 points)
2. Total overhead total: serial runtime = 15, Tp= parallel runtime= 5 , Tall = 5\*5= 25 -15= 10 minutes
3. Speedup Ts/P 15/5=3
4. Efficiency E= S/P: ⅗= .6
5. \*In this question, the computation is bounded by memory and performance is 1 FLOP / memory access. A processor with a cache yields an 70% hit ratio. If three processors are used, the hit ratio goes up to 80%. What is the Speedup if the cache access time is 1ns and DRAM access time is 100ns. Is it superlinear speedup? (10 points)

effective memory access time = 1ns x 0.7 + 100ns x 0.3 = 30.7 ns

processing rate = 1 / 30.7ns = 32.57 MFLOPS

effective memory access time = 1ns x 0.8 + 100ns x 0.2 = 20.8ns.

processing time in each processor = 1 / 20.8ns = 48.08 MFLOPS

total processing rate for 3 processors = 48.08 x 3 = 144.24 MFLOPS, and speedup is 144.24 / 32.57 = 4.43

not superliner speedup.