Filippo Zallocco

CS610 – 71429

Professor Turbé

11/28/2023

Report on a Matrix Multiplication on a Torus

Part 1

Using the Torus architecture to perform the matrix multiplication, we learn that all the data from matrices A and B is partitioned and assigned to a precise number of processors. In this case, 49 processors in total work on the values of the matrices, and the time to distribute the data is almost the same as that of process creation. We may appreciate from the values displayed in the terminal that Speedups remain consistent largely as we increase the communication delay-D and maintain the number of partitions constant. Naturally, the higher the number of partitions and the lower the communication delay, the more efficient the program’s performance will be. If we segment the program’s performance further in the interval from D1 = 200 to D4 = 100, we notice that the program’s acceleration gradually tilts downward. This pattern is repeated across all the intervals of p, from p = 2 to p = 5. The underlying reason for this trend is that in the far left of the diagram, the program spends more time creating processes than it does transferring data whereas the reverse occurs in the far right of the diagram despite increasing the communication delay as some processors have already carried out most of the instructions.

Part 2

Best performing test



This test has 20 for the Torus Communication Delay and 5 for the partition size.

Worst performing run



This run has 100 for the Torus Communication Delay and 2 for the partition size.