Jonathan Frame
PantarHei usr: sp19hpc_jmframe

CS 491/591: High Performance Computing Project 5
Parallel Programming with OpenMP Due date: 11:59 pm, April 21th, 2019

I have uploaded all source codes and a makefile as a tar file into Blackboard. Below are some results.

Part 1 (50 points) Matrix Multiplication
Change this to $N = 10^4$. You need to parallelize this matrix multiplication program in two different ways:

1. Add the necessary pragma to parallelize the outer for loop in the matrix multiplication;

Threads	Average calculation time N = 10,000
1	51.8
4	12.6
8	7.8
16	4.1
32	2.1

Note: To test the correctness of my matrix multiplication I set the values of matrices A and B to be equal to 1. Then each value of Matrix C should be equal to N.

2. Remove the pragma for the outer for loop and add the necessary pragma to parallelize the middle for loop in the matrix multiplication;

Threads	Average calculation time N = 10,000
1	50.5
4	7.6
8	3.9
16	2.0
32	0.9

Part 2 (20 points) Improve the provided code to avoid actually copying the array into another array by using a 3-dimensional array.

See source code. No deliverables were asked for this part other than code. Results shown below with Part 3 for comparison of sequential and multi-thread computations.

Part 3 (30 Points) Modify the sequential program in Part 2 to be an OpenMP program.

Again, see source code for Part 3 criteria.

I found that 5000 time steps was not sufficient to get assess the results of the calculation for a matrix with N=1000. The larger the matrix size the more time is required for convergence (see Courant–Friedrichs–Lewy condition). The results below are for a runtime (T) of 100,000.

The result at 100,000 time step:

20.0	20.0	20.0	100.0	100.0	100.0	20.0	20.0	20.0
20.0	23.4	34.3	53.4	59.7	53.4	34.3	23.4	20.0
20.0	22.5	27.2	33.0	35.5	33.0	27.2	22.5	20.0
20.0	21.0	22.5	24.1	24.8	24.1	22.5	21.0	20.0
20.0	20.3	20.6	21.0	21.1	21.0	20.6	20.3	20.0
20.0	20.0	20.1	20.2	20.2	20.2	20.1	20.0	20.0
20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0
20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0
20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0

The run time a speedup results are shown below:

Threads	Run Time (seconds)	Speedup
1	855	1
2	409	4
4	213	16
8	111	63
16	58	237
32	36	765