Name: Huayue Sun

Class: CS575

Professor: Mike Bailey

Project 2

1. Tell what machine you ran this on

Dell inspiron 15 5000 Core i5

x64 Window 10 home

Test the data using the CMD command

2. What do you think the actual volume is?

0.435308

3. Show the performances you achieved in tables and two graphs showing:

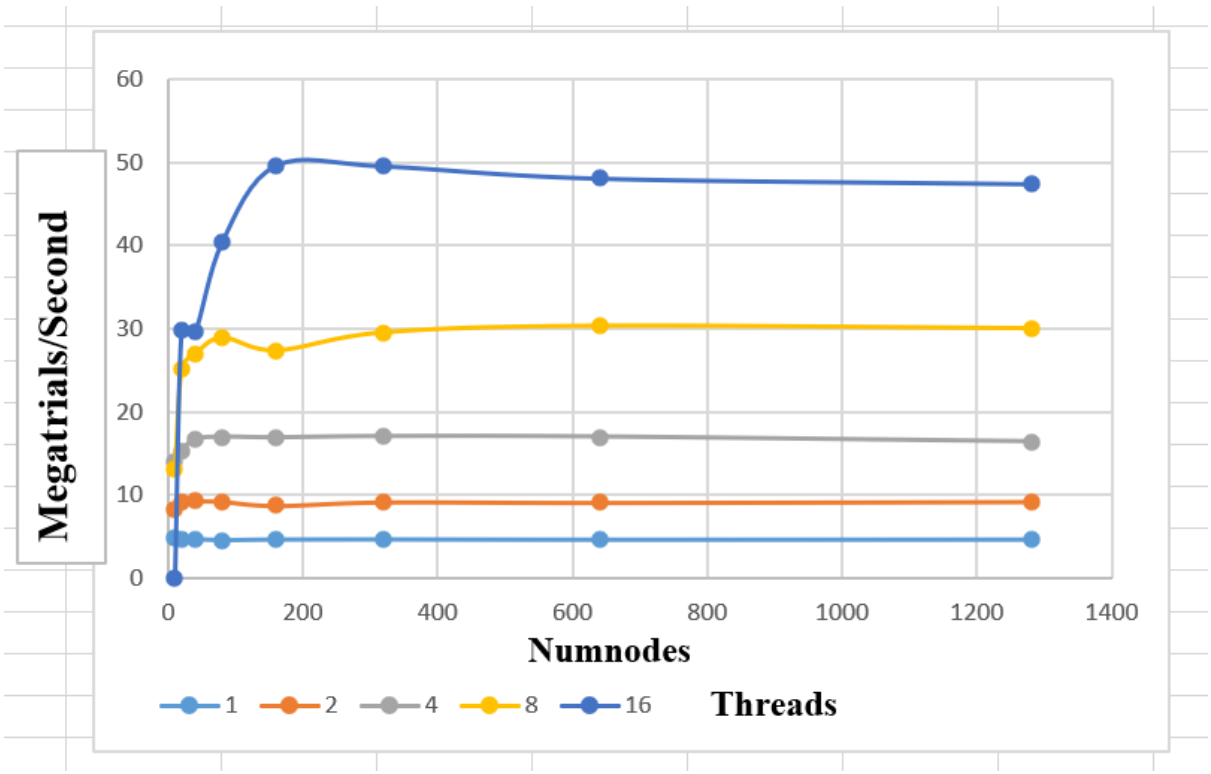
Performance as a function of NUMNODES with colored lines showing different NUMT values

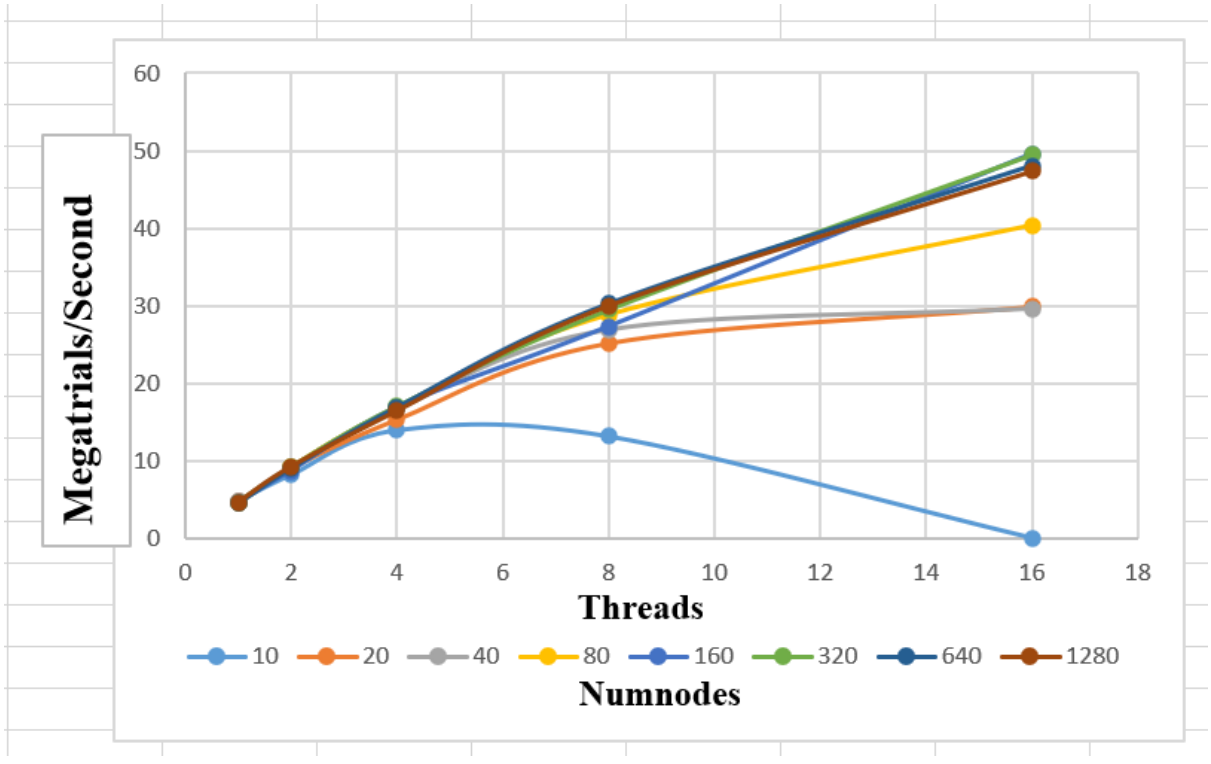
Performance as a function of NUMT with colored lines showing different NUMNODES values

(See the example in the Project Notes.)

Table 1

	10	20	40	80	160	320	640	1280
1	4.81	4.73	4.71	4.51	4.61	4.64	4.58	4.6
2	8.18	9.07	9.27	9.21	8.7	9.15	9.09	9.2
4	13.93	15.33	16.66	16.93	16.86	17	16.95	16.4
8	13.15	25.18	26.91	28.86	27.32	29.49	30.28	29.99
16	0.06	29.89	29.63	40.39	49.61	49.53	48.05	47.38





4. What patterns are you seeing in the speeds?

As can be seen from the chart, the increase or decrease of NUMNODES has little effect on the speed. Threads have a significant impact on speed, and data above 10 NUMNODES will

increase speed as the number of threads increases.

5. Why do you think it is behaving this way?

I think because multithreading speeds up the overall efficiency, but for each thread in multiple threads, it may not be as efficient as a single thread. Besides, I don't think the number of NUMNODES has much effect on program performance, indicating that the machine can load most of the different number of NUMNODES.

6. What is the Parallel Fraction for this application, using the Inverse Amdahl equation?

float Fp = (4 / 3) * (1 - (1 / S)).

So, first find S = (Performance with four threads) / (Performance with one thread)

=16.93 / 4.51

=3.75

So, Fp = (4 / 3) * (1 - (1 / 3.75)) = 0.98

7. Given that Parallel Fraction, what is the maximum speed-up you could ever get?

Maximum speed-up = 1 / (1 – Fp) = 1 / (1 – 0.98) = 50