Parallel Programming CS - 575

Project 1 - OpenMP: Monte Carlo Simulation

Submitted by

Rahul Kumar Nalubandhu

Onid : nalubanr

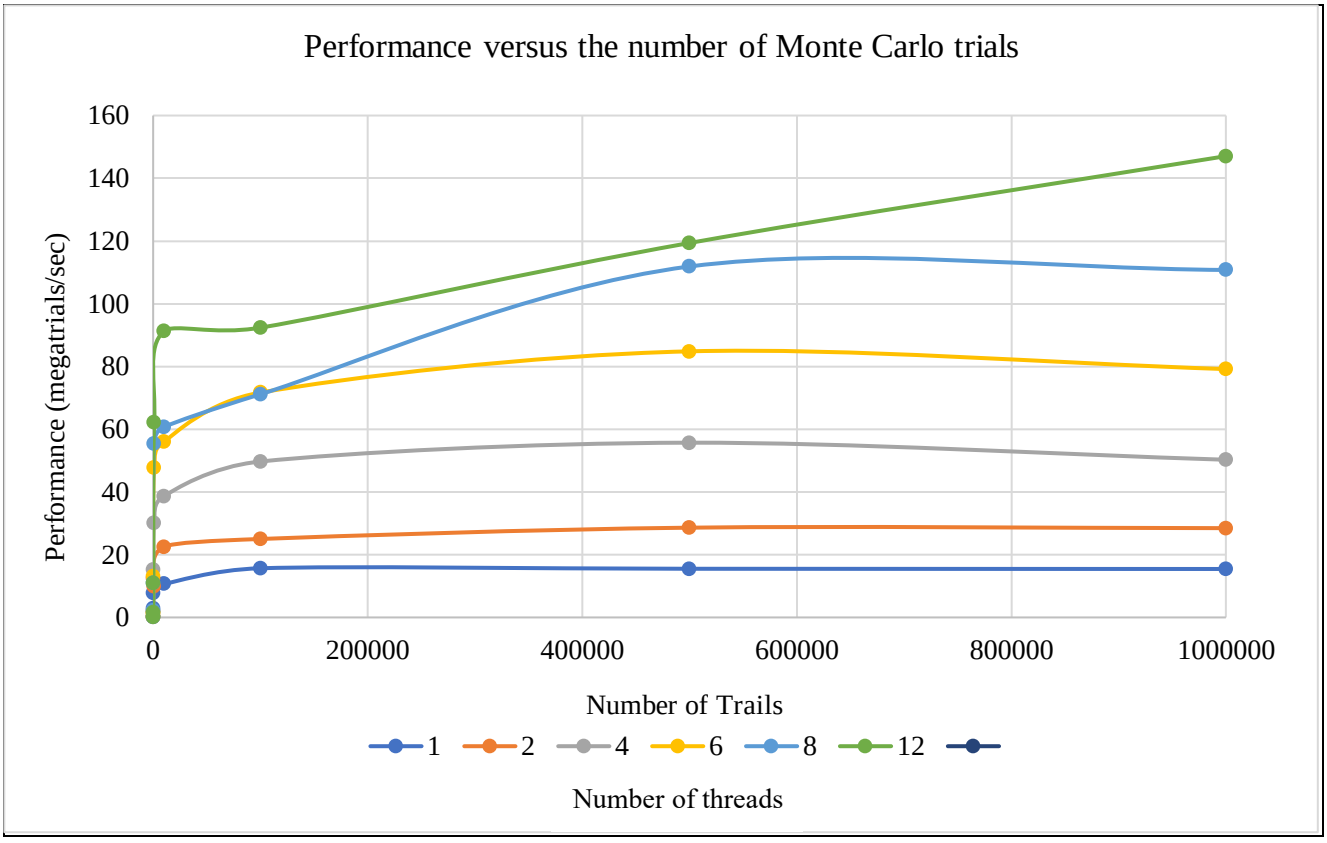
Email : nalubanr@oregonstate.edu

1) A rectangular data table of the performance numbers as a function of threads and NUMTRIALS.
For this I have taken (1, 2, 4, 6, 8, 12) threads and Trails (1, 10, 100, 1000, 10000, 100000, 500000, 1000000).

Threads/Trials	1	10	100	1000	10000	100000	500000	1000000
1	0.37	2.91	7.7	11.25	10.69	15.68	15.52	15.45
2	0.22	1.63	12.54	9.85	22.51	25.01	28.62	28.46
4	0.24	2.06	15.15	30.18	38.67	49.71	55.71	50.27
6	0.25	1.51	13.12	47.68	56	71.68	84.83	79.17
8	0.18	2.19	11.01	55.36	60.69	71.15	111.88	110.77
12	0.17	1.53	11.08	62.24	91.22	92.37	119.37	147.05

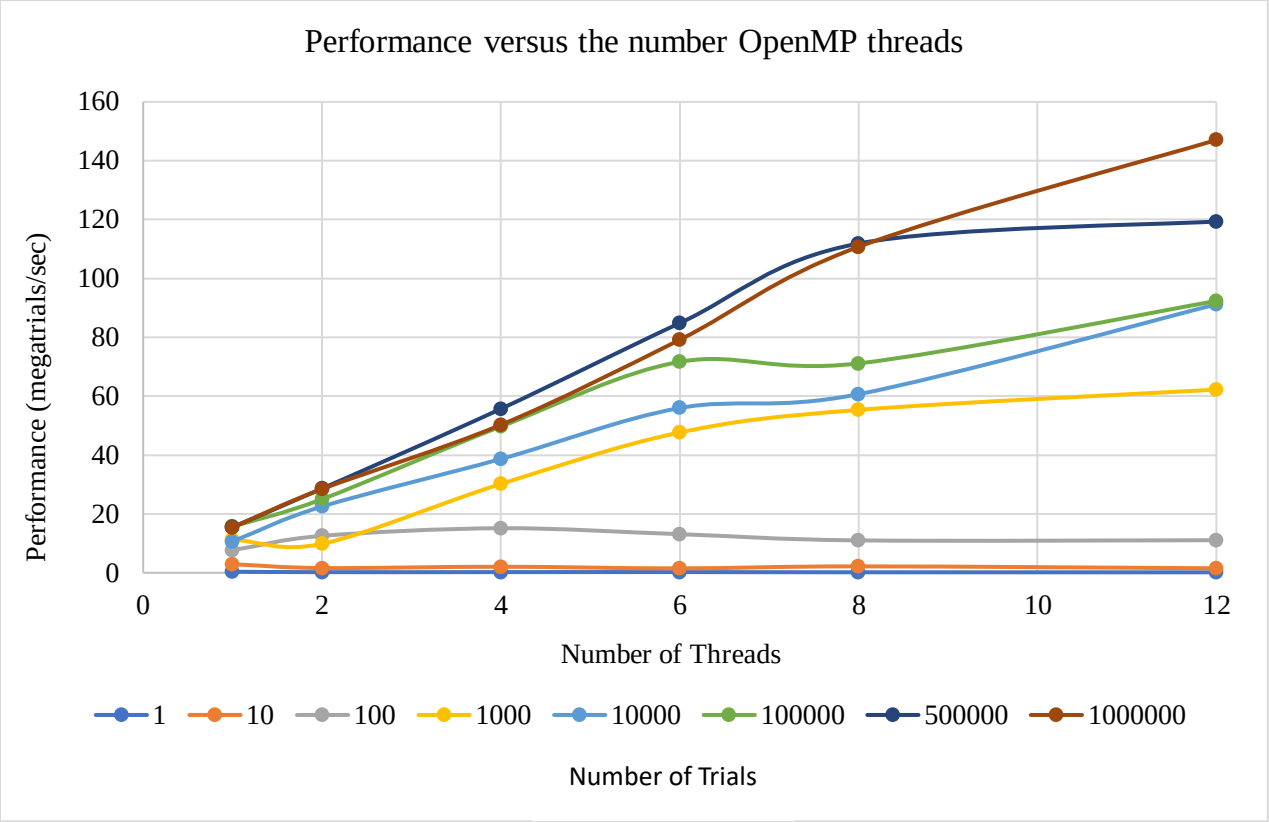
2) The **2** performance graphs. The two graphs need to be:

1) Performance versus the number of Monte Carlo trials, with the colored lines being the number of OpenMP threads.



Graph1

2) Performance versus the number OpenMP threads, with the colored lines being the number of Monte Carlo trials.



Graph2

3) Your estimate of the Probability.

Choosing one of the runs (one of the ones with the maximum number of trials would be good), tell me what you think the actual probability is.

The actual **probability is 26.82%.** This is by selecting the probabilities of thread 12 for maximum number of trails i.e., 1000000, hence the probability of pins fitting into the hole is 26.82%.

Threads	Trials	Probability	megatrials/sec
1	100000	26.82	15.68
1	500000	26.8	15.52
1	1000000	26.95	15.45
2	100000	26.94	25.01

2	500000	26.95	28.62
2	1000000	26.88	28.46
4	100000	26.71	49.71
4	500000	26.87	55.71
4	1000000	26.78	50.27
6	100000	27.06	71.68
6	500000	27.01	84.83
6	1000000	26.89	79.17
8	100000	26.84	71.15
8	500000	26.86	111.88
8	1000000	26.87	110.77
12	100000	26.89	92.37
12	500000	26.76	119.37
12	1000000	26.82	147.05

4) Your estimate of the Parallel Fraction (*show your work!*).

For 1 thread: 1000000 trials ; probability = 26.81 ; megatrials/sec = 17.77

For 12 threads : 1000000 trials ; probability = 26.87 ; megatrials/sec = 172.90

$$\text{Speedup(S)} = \frac{\text{Performance of 12 threads (Maximum)}}{\text{Performance of 1 thread}}$$

S = 172.90 / 17.77

Speedup = 9.729

Fp = (n / n-1) * (1 – (1 / S)). n => max number of threads = 12

Fp = (12 / 11) * (1 – (1 / 9.729))

Fp = 0.97

Parallel Fraction (Fp) = 0.97

5) Your commentary: why do the graphs look the way they do? What are they telling you?

In graph 1 which demonstrates the relationship between performance vs number of trails, we can see that in the initial stage the performance is very slow because of the amount of work done in the initial stage is less and the number of trails is less. Also, we can see that maximum threads have high performance which demonstrated the benefits of increased parallelism.

In Graph 2, which demonstrates the relationship between performance and the number of threads, we can observe a positive correlation between the two variables. As the number of threads increases, the performance also improves. This is because an increase in parallelism allows for more efficient handling of larger datasets, resulting in enhanced performance.