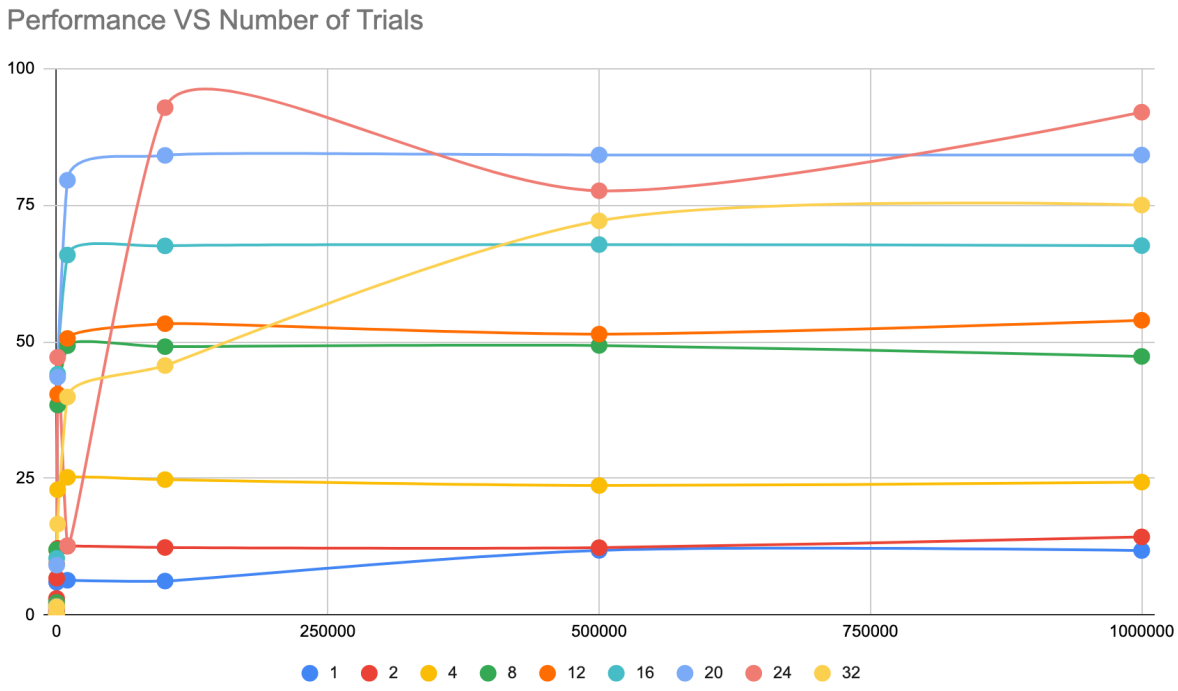
Alexis Doyle
doylalex@oregonstate.edu

Project 01

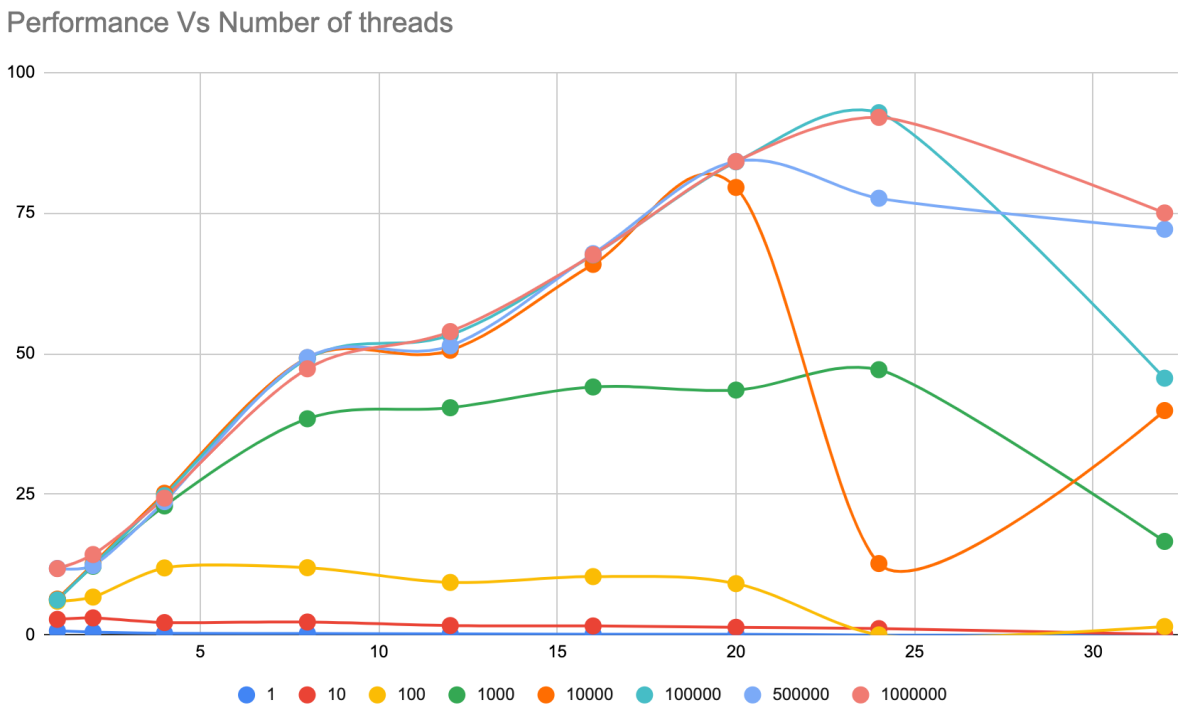
1. *A rectangular data table of the performance numbers as a function of threads and NUMTRIALS.*

	1	10	100	1000	10000	100000	500000	1000000
1	0.74	2.79	6.01	6.34	6.38	6.23	11.84	11.81
2	0.54	3.05	6.75	12.22	12.62	12.37	12.34	14.31
4	0.32	2.22	11.93	22.94	25.2	24.78	23.7	24.33
8	0.27	2.32	11.97	38.43	49.28	49.1	49.32	47.31
12	0.2	1.69	9.35	40.39	50.6	53.3	51.39	53.92
16	0.15	1.62	10.39	44.08	65.86	67.53	67.77	67.58
20	0.15	1.38	9.14	43.53	79.53	84.08	84.16	84.14
24	0	1.15	0.03	47.15	12.71	92.81	77.61	91.98
32	0.01	0.12	1.51	16.64	39.89	45.64	72.1	75

2. *The 2 performance graphs. The two graphs need to be:*
   a. *Performance versus the number of Monte Carlo trials, with the colored lines being the number of OpenMP threads.*



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



3. *Your estimate of the Probability.*

The Probability I am choosing is the run that is 32 threads with 1,000,000 Trials and has 75 MegaTrials/Sec. The probability calculated is 26.87. I chose this probability because looking at a lot of different runs most of the probabilities are around 26 or 27 so it seemed like the most accurate probability.

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

1 thread with 1,000,000 Trials and has 11.81 MegaTrials/Sec.

32 threads with 1,000,000 Trials and has 75 MegaTrials/Sec.

Speed up = peak performance for 32 / peak performance for 1 = 6.350550381

Parallel fraction = n/(n-1) * (T1 - Tn)/T1 or n/(n-1) * (1-(1/speedup)
= 32/(32-1) * (1-(1-6.350550381) = 0.869711828

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

Performance vs trials:

The graph seems to plateau around 10,000 trials and stay pretty consistent with performance for most of the threads. This graph can lead one to believe that when increasing the number of trials after a while your performance will start to plateau and not improve when using a fixed amount of threads.

Performance vs threads:

Overall the graph's performance seems to increase with the number of threads increasing. The amount of threads do not seem to matter as much for trials that are less than 100. Trials that are over 100 seem to have a more obvious increase in performance with more threads. This graph shows a correlation between the number of trials and threads. If you are going to have more trials than you want more threads. However around 10,000 trials the performance in all except one thread seemed to go down. Meaning there is a limit to the positive effect more threads can have when processing lots of trials.

Both:

In both graphs the data pattern shifted after reaching 10,000 threads. For the trial graph it went from increasing to staying consistent and the thread graph went from increasing to decreasing.