Project 1: Monte Carlo Simulation

Joel Huffman huffmajo@oregonstate.edu

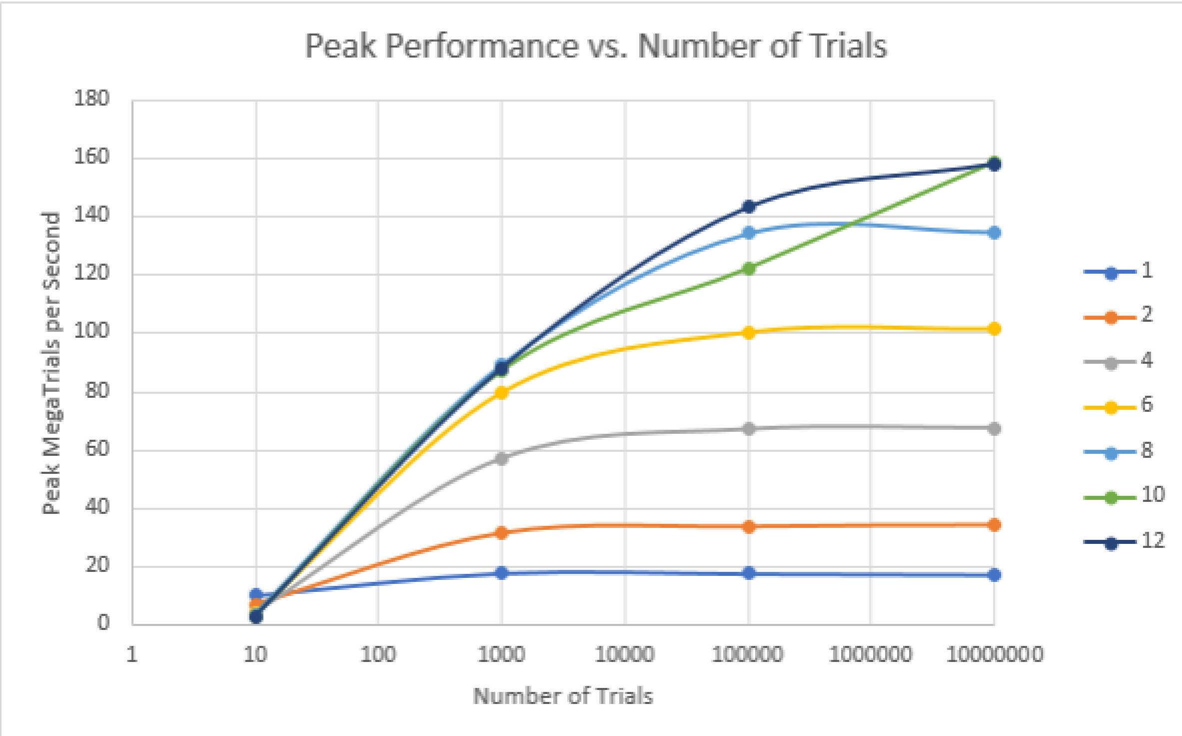
Tables and Graphs

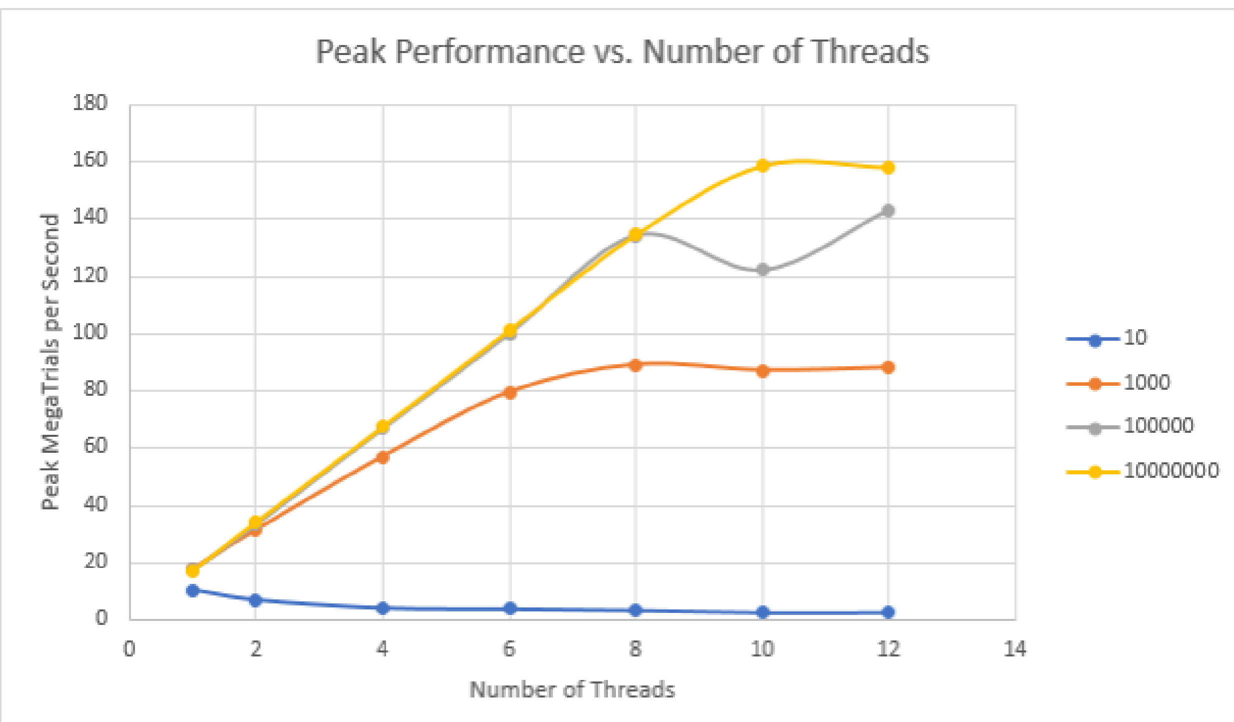
Peak Performance Across Number of Trials and Number of Threads

		Number of Trials			
		10	**1000**	**100000**	**10000000**
Number of Threads	**1**	10.24685	17.80818	17.64512	17.293013
	2	6.992783	31.49103	33.72904	34.260624
	4	4.353522	57.07446	67.09135	67.535927
	6	3.987159	79.63081	100.1878	101.31315
	8	3.535971	89.16641	134.3709	134.75348
	10	2.755569	87.16145	122.2687	158.82726
	12	2.790208	88.12901	143.2854	158.22583

Probability of Hitting Plate Across Number of Trials and Number of Threads

		Number of Trials			
		10	**1000**	**100000**	**10000000**
Number of Threads	**1**	0.1	0.784	0.19073	0.190309
	2	0.2	0.194	0.19007	0.190347
	4	0.4	0.202	0.18977	0.190325
	6	0.2	0.19	0.18887	0.190156
	8	0.1	0.182	0.18948	0.190403
	10	0.1	0.187	0.19044	0.190343
	12	0.1	0.195	0.19055	0.190303


Peak Performance vs. Number of Trials
Peak MegaTrials per Second
Number of Trials
0
20
40
60
80
100
120
140
160
180
1
10
100
1000
10000
100000
1000000
10000000
1
2
4
6
8
10
12


Peak Performance vs. Number of Threads
Peak MegaTrials per Second
Number of Threads
0
20
40
60
80
100
120
140
160
180
0
2
4
6
8
10
12
14
10
1000
100000
10000000

Explanation and Analysis

All of the runs using 10,000,000 trials yielded around 0.19 as the probability of the beam striking the plate. These consistent results across all tested thread counts with higher numbers of trials lead me to believe the actual probability is 0.19.

Speedups for each of the thread counts are calculated and shown below. The peak performance values were all pulled from the runs using 10,000,000 trials. Speedup efficiency decreases as the thread count increases and seems to start to hit a limit around 12 threads. We may be approaching the optimal performance of the parallel portion of the program around these thread counts.

$S_2 = P_2 / P_1 = 34.260624 / 17.293013 = 1.98$
$S_4 = P_4 / P_1 = 67.535927 / 17.293013 = 3.91$
$S_6 = P_6 / P_1 = 101.313148 / 17.293013 = 5.86$
$S_8 = P_8 / P_1 = 134.753478 / 17.293013 = 7.79$
$S_{10} = P_{10} / P_1 = 158.827255 / 17.293013 = 9.18$
$S_{12} = P_{12} / P_1 = 158.22583 / 17.293013 = 9.15$

The Parallel Fraction values are calculated and shown below for each thread count. Fp only just starts to dip around the 12 thread end of our testing. This may show that we maybe be approaching our max speedup soon after 12 threads.

$Fp_2 = \frac{2}{1}\left(1 - \frac{1}{1.98}\right) = 0.990$
$Fp_4 = \frac{4}{3}\left(1 - \frac{1}{3.91}\right) = 0.992$
$Fp_6 = \frac{6}{5}\left(1 - \frac{1}{5.86}\right) = 0.995$
$Fp_8 = \frac{8}{7}\left(1 - \frac{1}{7.79}\right) = 0.996$
$Fp_{10} = \frac{10}{9}\left(1 - \frac{1}{9.18}\right) = 0.990$
$Fp_{12} = \frac{12}{11}\left(1 - \frac{1}{9.15}\right) = 0.972$