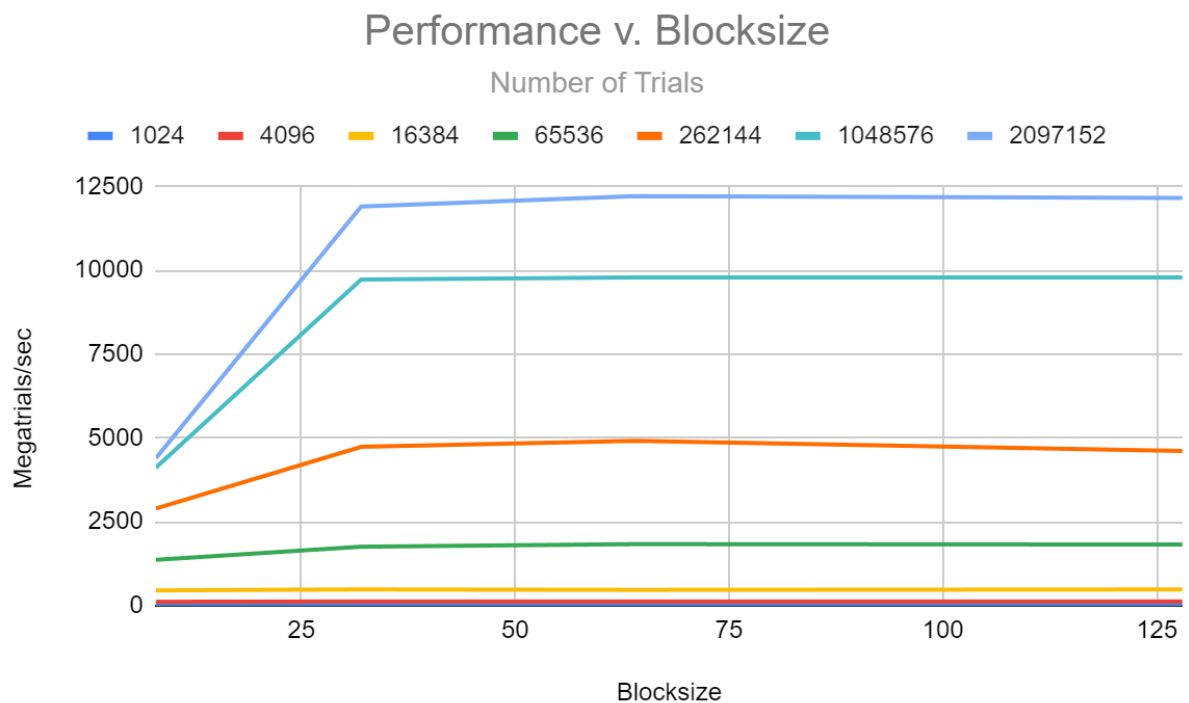
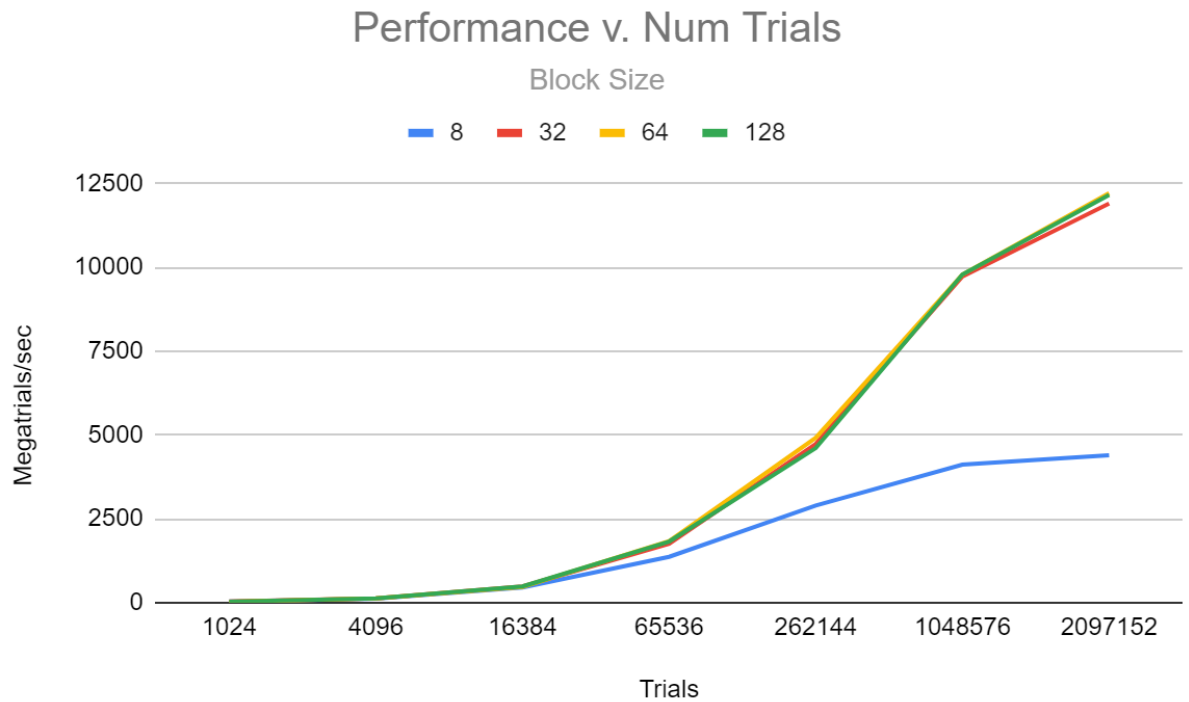


CS 475 Parallel Programming: Project 5 - CUDA Monte Carlo

Conner Rhea
rheac@oregonstate.edu

1. I ran this code on the DGX system at OSU by SSH-ing into it.
2. The new probability appears to be sitting pretty at 74.6% for the Monte Carlo simulation, it consistently hit around that range.
3. Table and Charts:

Trials	Blocksize	Performance	Probability
1024	8	32.2581	74.61
1024	32	32.2581	73.93
1024	64	31.25	75.29
1024	128	31.25	71.97
4096	8	117.6471	75.1
4096	32	125	74.44
4096	64	125	74.19
4096	128	125	74.73
16384	8	457.1428	74.54
16384	32	484.8485	74.94
16384	64	470.5882	74.78
16384	128	484.8485	74.35
65536	8	1368.0695	74.53
65536	32	1759.4501	74.64
65536	64	1836.7713	74.44
65536	128	1823.6865	74.6
262144	8	2898.7968	74.74
262144	32	4735.2601	74.74
262144	64	4920.1202	74.71
262144	128	4612.6126	74.7
1048576	8	4119.1704	74.67
1048576	32	9729.2161	74.68
1048576	64	9787.3354	74.66
1048576	128	9790.2596	74.67
2097152	8	4401.0475	74.71
2097152	32	11906.9767	74.69
2097152	64	12213.1945	74.72
2097152	128	12163.3258	74.69



- Performance sharply increases as Blocksize increases up to at least 32, then begins to level off and stay relatively constant across the trial sizes.
- As described in the lecture, full compute power for GPU's scales upwards cleanly. Even into the millions GPU computing should be getting full results,

in fact even better results than with the smaller data set, which can explain the increase in performance even in the BLOCKSIZE 8 in Performance v. NumTrials

6. A BLOCKSIZE of 8 does not constitute a full Warp in CUDA, so only having 8 Blocks available means there will be some overhead waiting for data to be finished in memory.
7. In Project 1 performance had a significant dip as the trial size began to get really large, however that is not so the case here, where performance actually continued to increase as the data set got very big. Since GPU computing scales upwards well, this makes sense that the performance gains with CUDA did not fall off.
8. GPU Parallel Computing seems like an excellent way to handle extremely large data sets that scale upwards due to providing performance increases as data sets reach up into the millions. For simple computing such as this, it provides incredible gains.