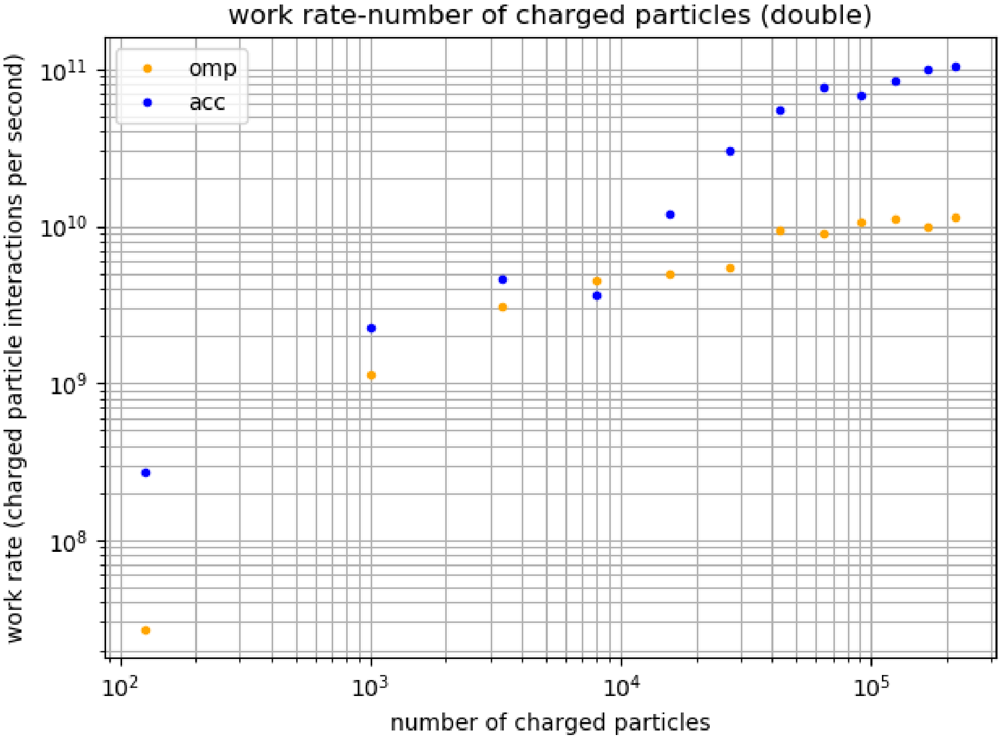
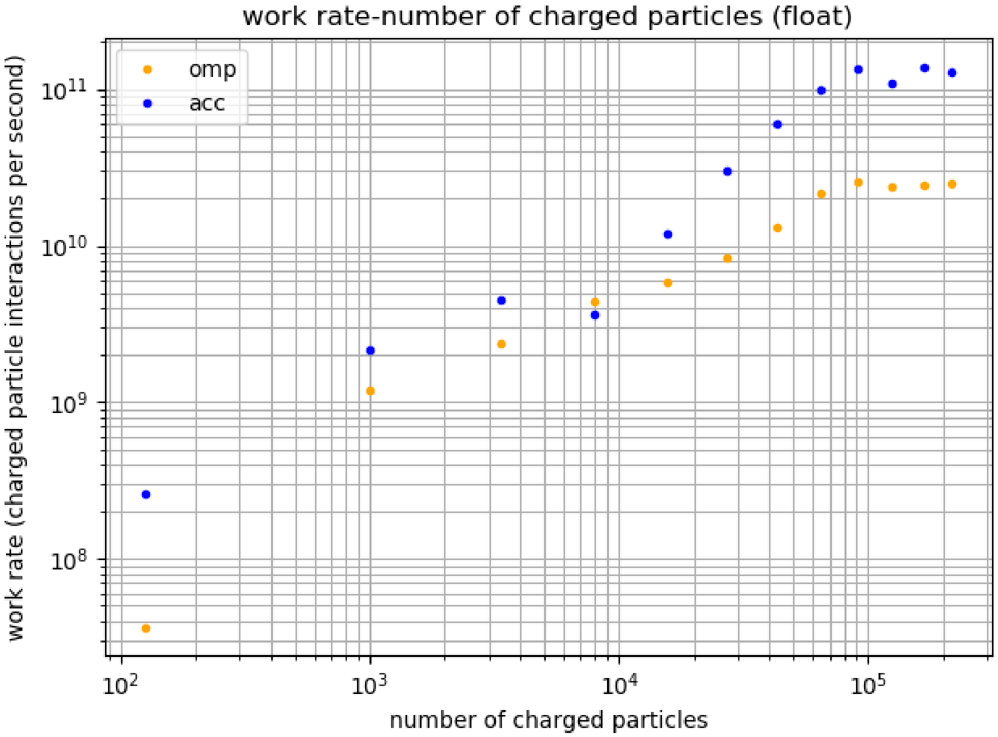
Exercise 3: n-body charged particle simulation

Tian-Ruei Kuan & Ting-Yu Lan





Floating point operations per charged particle interactions (count sqrt as 1) is 16.
Let n = the number of charged particles

Arithmetic intensity

Total flops = $16n^2$
Data read: $6 \times 8n$ bytes
Data written: $6 \times 8n$ bytes
Total data moved: Read + Write = $96n$ bytes
Arithmetic intensity: $\frac{16n^2}{96n} = \frac{1}{6}n$ flops/byte
Similarly, for float, the arithmetic intensity is $\frac{1}{3}n$.

CPU

Machine balance of CPU:

Assume the flop rate for coc-ice is the same for coc-ice-gpu
R_{peak}= 1.5 TFlop/s for double, 2.8 TFlop/s for single
Memory bandwidth for L3 cache (reference) is about 1000 GB/s for 12 cores, that is 2000 GB/s for 24 cores. Because we set OMP_PROC_BIND to sperate, the 24 cores will be assigned into 2 sockets with 12 cores each.
$B_{L3} = 2000\ GB/s$
Machine balance = $\frac{R_{peak}}{B_{L3}} = 0.75\ Flop/Byte$ for double, $1.4\ Flop/Byte$ for single

Arithmetic intensity > Machine balance when $1/6\,n > 1.4 \rightarrow n > 8$ (for both float and double). For all the data points in the chart, n > 8. Therefore, it is always computationally intensive, and the idea flop rate is always R_{peak} = 1.5 TFlop/s for double and 2.8 TFlop/s for single.

Total time of CPU:

For double, total time = $\frac{16n^2}{1.5T}\ s = 1.07n^2 \times 10^{-11}\ s$
For single, total time = $\frac{16n^2}{2.8T}\ s = 5.7n^2 \times 10^{-12}\ s$

GPU

Machine balance of GPU (reference):

R_{peak} = 7 TFlop/s for double, 14TFlop/s for single
Memory Bandwidth: B = 900 GB/s
Machine balance = $\frac{R_{peak}}{B} = 7.8\ Flop/Byte$ for single, $15.6\ Flop/Byte$ for double

Arithmetic intensity > Machine balance when $1/6\,n\ >\ 15.6\ \rightarrow n > 93$ (for both float and double). For all the data points in the chart, min n = 125 > 93. Therefore, it is always computationally intensive, and the idea flop rate is always R_{peak} = 7 TFlop/s for double and 14TFlop/s for single.

Data transfer time of GPU (reference):

Interconnect Bandwidth: 32 GB/s
Total data size: $96n$ bytes
Data transfer time =$\frac{96n\ byte}{32GB} = 3\ ns$

Total time of GPU:

For double, total time = $\frac{16n^2}{7T} + 3 \times 10^{-9}\ s = \ \frac{16n^2}{7}n^2 \times 10^{-12} + 3 \times 10^{-9}\ s$
$= 2.3n^2 \times 10^{-12} + 3 \times 10^{-9}\ s$
For single, total time = $\frac{16n^2}{14T} + 3 \times 10^{-9}\ s = \ \frac{8n^2}{7}n^2 \times 10^{-12} + 3 \times 10^{-9}\ s$
$= 1.1n^2 \times 10^{-12} + 3 \times 10^{-9}\ s$

Comparison between CPU and GPU

For both double and single, GPU is faster than CPU when n is large by looking at the coefficient of the n^2 term. When n is small (the smallest n in the chart is 125), GPU is still faster.

For the 4th point (num charges = 8000, n_steps = 10) in the chart, GPU is slower than CPU. This is because we do not parallelize the initialization part (state.cc) for GPU, but it is parallelized for CPU. This become significant when the total time is small, which is the case for the 4th point in the chart.