Exercise 3: n-body charged particle simulation

Tian-Ruei Kuan & Ting-Yu Lan

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Floating point operations per charged particle interactions (count sqrt as 1) is 16.

Let n = the number of charged particles

Arithmetic intensity

Total flops =

Data read: bytes

Data written: bytes

Total data moved: Read + Write = bytes

Arithmetic intensity: flops/byte

Similarly, for float, the arithmetic intensity is .

CPU

**Machine balance of CPU:**

Assume the flop rate for coc-ice is the same for coc-ice-gpu

= 1.5 TFlop/s for double, 2.8 TFlop/s for single

Memory bandwidth for L3 cache ([reference](https://rrze-hpc.github.io/INSPECT/machines/CascadelakeSP_Gold-6248_SNC?fbclid=IwAR243x80DKCYUWw_O9S534Ux_wV5YrrOiksCMuRf2vjzYFJYBj6e6E3C55o)) 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.

Machine balance = for double, for single

Arithmetic intensity > Machine balance when (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 = 1.5 TFlop/s for double and 2.8 TFlop/s for single.

**Total time of CPU:**

For double, total time =

For single, total time =

GPU

**Machine balance of GPU (**[**reference**](https://images.nvidia.com/content/technologies/volta/pdf/volta-v100-datasheet-update-us-1165301-r5.pdf)**):**

= 7 TFlop/s for double, 14TFlop/s for single

Memory Bandwidth: B = 900 GB/s

Machine balance = for single, for double

Arithmetic intensity > Machine balance when (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 = 7 TFlop/s for double and 14TFlop/s for single.

**Data transfer time of GPU (**[**reference**](https://images.nvidia.com/content/technologies/volta/pdf/volta-v100-datasheet-update-us-1165301-r5.pdf)**):**

Interconnect Bandwidth: 32 GB/s

Total data size: bytes

Data transfer time =

**Total time of GPU:**

For double, total time =

For single, total time =

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 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.