APMA 2822b Homework 1

Ryan Greenblatt

February 2019

2

The CPU has a peak FLOP rate of 500 GFLOPS[1] and an achievable bandwidth of 100 GiB/s. Thus, memory will be the bottleneck unless the task has an operational intensity of 5 FLOPS per byte or higher.

a

For task a, each iteration requires 3 FLOPS (the compiler may optimize it to just 2 FLOPS) and 3 doubles to be transfered. Each double is 8 bytes, so the FLOPS per byte is only $\frac{3}{24} = \frac{1}{8} = 0.125$. The task will be memory limited and the total amount of memory transfer is $\frac{512*1024*1024}{8} * 8 * 3$ which is 1.5 GiB. It will take $\frac{3}{200} = 0.015$ seconds. The total number of FLOPS is $\frac{512*1024*1024}{8} * 3$ and the FLOP rate will be $\frac{25}{2} = 12.5$ GFLOPS.

b

For task b, each iteration requires 6 FLOPS and 5 doubles. Each double is 8 bytes, so the FLOPS per byte is only $\frac{3}{20} = 0.15$. The task will be memory limited and the total amount of memory transfer is $\frac{512*1024*1024}{8} * 8 * 5$ which is 2.5 GiB. It will take $\frac{5}{200} = \frac{1}{40} = 0.025$ seconds. The total number of FLOPS is $\frac{512*1024*1024}{8} * 6$ and the FLOP rate will be 15 GFLOPS.

3

I ran my code on the CCV using the class reserved node. Both tasks were memory limited for both sizes. Running the tasks with reduced size increased speed substantially because the data could remain entirely in the cache.

Task	N	Seconds	GiB/s
a	$\frac{512*1024*1024}{8}$	0.0368	27
b	$\frac{512*1024*1024}{8}$	0.0651	23
a	1024	1.70e-07	90
b	1024	3.52e-07	65

[1] Within this report, I am assuming 1 GFLOPS is 1024^3 FLOPS.