Programming assignment 1

Anusha Ravilla

200108828

1/26/2016

2.

a)

A Matrix Size = 2048 X 2048. H Matrix Size = 7 X 7

Two operations are performed by every thread (Addition and Subtraction). Therefore the number of floating point operations per Second = 2*7*7 (the loop is for H matrix width iterations, max(H dimension) = 7).

Two accesses to memory per iterations (A and B matrices) . Kernel Code has two loops with the width and height of H matrix. Size double is used in the algorithm (size is 8 bytes for double).

Number of memory access read operations = 2*7*7*8

Arithmetic Intensity = $\frac{Number\ of\ Computations\ performed}{Number\ of\ accesses\ to\ memory(bytes)}$ = $\frac{2*7*7}{2*7*7*8}$ = 1/8 = 0.125

b)

According to the fig. 4.11, the performance is bound by the memory bandwidth for value of AI of 1/8 (0.125) on Intel Core i7 920. Hence, it is a memory bandwidth problem.

c)

Arithmetic Intensity (Number of floating-point operations per byte read) from the convolution algorithm:

Arithmetic Intensity = $\frac{Number\ of\ Computations\ performed}{Number\ of\ accesses\ to\ memory(bytes)}$ = $\frac{(2048+7-1)*(2048+7-1)*2*7*7}{((7*7)+(2048*2048))*8}$ = 12.25

$((7*7)+(2048*2048))*8$ -> 8 (size of double - assuming double), The algorithm accesses memory so as to get 7*7 and 2048*2048 accesses. So AI is 12.25

d) According to the fig. 4.11, the performance is bound by peak floating point computation value for value of AI of 12.25 on Intel Core i7 920. Hence, it is a memory computational performance problem.

3.

For the code, the time for execution of the kernel takes 5 micro seconds for the given reference input 5X5 matrix and the convolution kernel given. After performing an optimization on the shared memory access to the convolution kernel, the execution of the kernel takes 4 micro seconds on an average. Hence we can see an improvement in the performance by utilizing shared memory.