EE147
Lab 2 Report: Tiled Matrix Multiplication

Victoria Hall, 861154075

April 27, 2018

Questions

1. In your kernel implementation, how many threads can be simultaneously executing? Assume a GPU which has 30 streaming multiprocessors.

   For 1 SM processing 1 warp of 32 threads, we would get:
   30*32 = 960 threads executing simultaneously.
   This number could be multiplied if each SM is capable of processing more than 1 warp at a time.

2. Use nvcc –ptxas-options=”-v” to report the resource usage of your implementation. Note that the compilation will fail but you will still get a report of the relevant information. Experiment with the Nvidia visual profiler, which is part of the CUDA toolkit, and use it to further understand the resource usage. In particular, report your branch divergence behavior and whether your memory accesses are coalesced.

   Output of nvcc –ptxas-options=”-v”: Used 25 registers, 2048 bytes smem, 360 bytes cmem[0].

   According to the Nvidia visual profiler, there are 0 branch divergences. I thought there would be many branch divergences at the edges of the matrices since the 1000x1000 default case is not easily divisible by the 16x16 tile size we are using. This could be that the Nvidia visual profiler may only consider a branch divergent if one branch does something and another branch does something else but it doesn’t consider it divergent if the second branch just does nothing.

   For the memory, our memory accesses are not coalesced for matrices bigger than the 16x16 tile. The 16x16 tile will have threads in a warp that are not sequential in indexing.

3. How many times is each element of the input matrices loaded during the execution of the kernel?

   For the default function running matrices of 1000x1000, the number of global loads is 4,194,304. This would be about 2 loads per element for the two matrices we are multiplying. In this same case, the number of shared loads is 40,320,000. This would be about 20 loads per element. Overall, for both the global and shared loads, the kernel loads each element approximately 22 times.