EC527 Lab 8: Matrix Multiply on GPU

John Burke

April 13, 2017

1 Comparison of Methods Implemented

Input Length	1k	2k
CPU	13,727	249,577
GPU Global		
Kernel	228.9	2148.9
Kernel and Transfer	233.0	2160.5
GPU Shared		
Kernel	11.9	95.3
Kernel and Transfer	16.5	109.5
GPU Unrolled (by 4)		
Kernel	3.1	24.6
Kernel and Transfer	7.3	38.1

Short Analysis of Results The above table lists the time in `ms` for the execution of the various implementations of matrix multiply. As can be seen, GPU implementations over both sizes, one's that both fit and don't fit in the CPU cache systems, are advantageous to the CPU designs. Taking further advantage of the GPU by making smart use of shared memory and getting more work per threads in a 16 by 16 block to better fill a given warp also improve the time it takes to complete the matrix multiplication.

Notes on Code Included All the code can be found in the code directory included. Each implementation of the GPU Matrix Multiply is in a different source file that also compares it against CPU baseline code. The global memory only code is in `global_mmm.cu`, the one that makes use of shared memory is in `shared_mmm.cu`, and finally the one that makes use of shared memory and unrolling is in `shared_unroll_mmm.cu`.