

Jung-Che Chang

Project #6

OpenCL Matrix Multiplication

changju@oregonstate.edu

1. What machine you ran this on

DGX system

2. Show the table and graphs

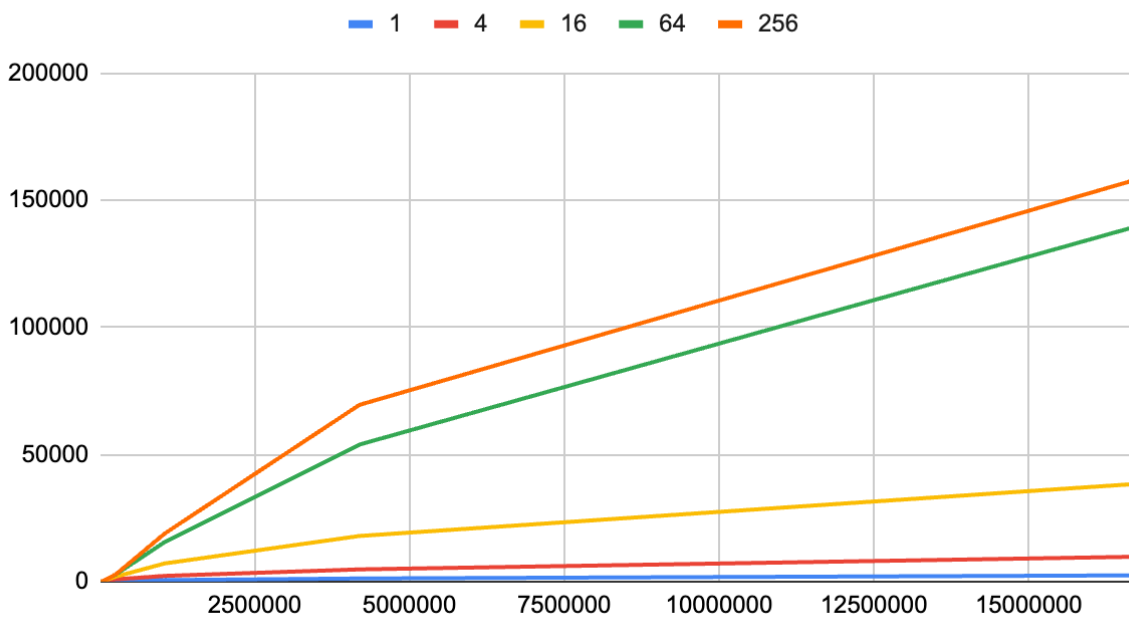
Matrix Size	Work Elements	GigaMultsPerSecond
16384	1	28
16384	4	42.8
16384	16	48.33
16384	64	49.58
16384	256	49.55
65536	1	111.71
65536	4	214.85
65536	16	350.02
65536	64	386.54
65536	256	397.62
262144	1	281.36
262144	4	893.82
262144	16	1968.89
262144	64	2812.25
262144	256	2892.86
1048576	1	602.25
1048576	4	2248.51
1048576	16	7143.12
1048576	64	15533.32
1048576	256	18903.45

4194304	1	1225.59
4194304	4	4808.45
4194304	16	17950.52
4194304	64	53906.01
4194304	256	69535.57
16777216	1	2462.58
16777216	4	9798.95
16777216	16	38490.27
16777216	64	139989.6
16777216	256	158432.95

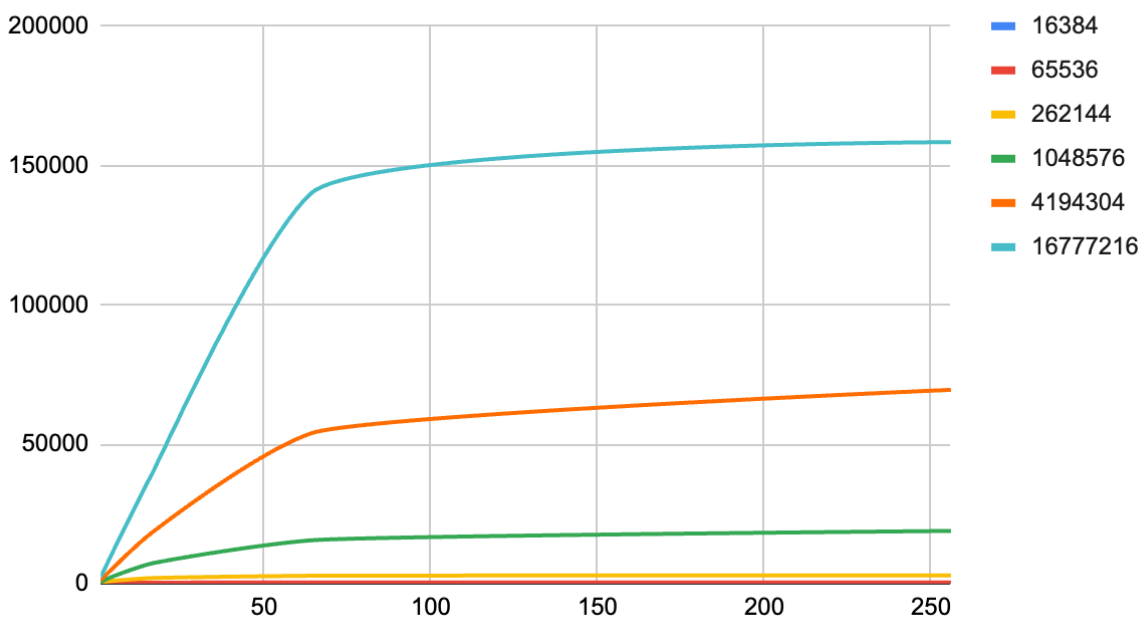
Pivot Table:

<i>GigaMultsPerSecond的SUM</i>	<i>Work Elements</i>				
<i>Matrix Size</i>	1	4	16	64	256
16384	28	42.8	48.33	49.58	49.55
65536	111.71	214.85	350.02	386.54	397.62
262144	281.36	893.82	1968.89	2812.25	2892.86
1048576	602.25	2248.51	7143.12	15533.32	18903.45
4194304	1225.59	4808.45	17950.52	53906.01	69535.57
16777216	2462.58	9798.95	38490.27	139989.6	158432.95

Matrix multiply performance versus total matrix size



Matrix multiply performance versus total Local Size



3. What patterns are you seeing in the performance curves? What difference does the size of the matrices make? What difference does the size of each work-group make?

For a given total matrix size, the performances increase when the local work size increase. In the case of the local size, the performance increases when the total matrix size increase.

4. Why do you think the patterns look this way?

When the local work size is limited (ranging from 1 to 16), numerous processing elements within the compute units remain inactive, resulting in a substantial waste of computational time. Similarly, if the global dataset size is overly small, the GPU isn't adequately utilized and the amount of work executed on the GPU fails to offset the overheads involved in the initial setup.