Report 8
OpenCL.

Paweł Lipiór, Michał Żoczek

January 2, 2020

1 Laboratory classes

During this laboratory classes we have learned another method of parallel computing. This time we have used OpenCL framework. Classes were maintaily about understanding examples of OpenCL code and compare it to CUDA solutions. Our practical exercise was to write code in OpenCl to add multiple vectors. Next step was compare timing of execution to corresponding code written in CUDA.

2 Performance tests

After code all necessary functionalities we were ready to test performance. Using `<sys/time.h>` library we were able to get time of execution for both programs. We decided to count time from the vector initialization part and end after kernel execution. We decided to test the performance for a vector with length with successive powers of two. All collected data are shown in pictures 1, 2, 3:

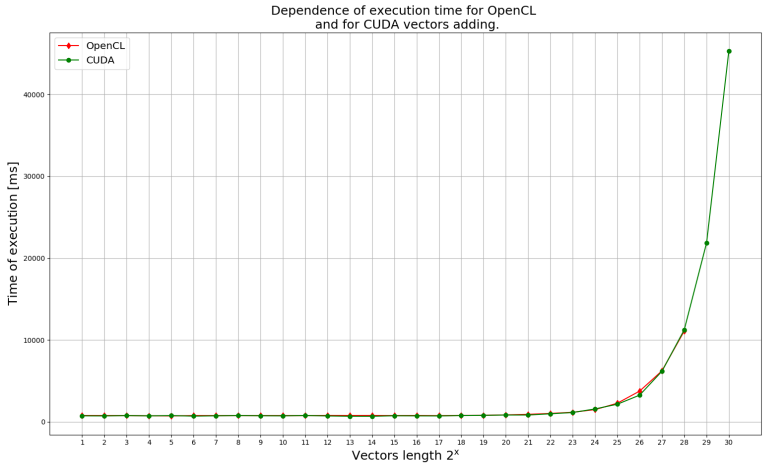

Figure 1: Dependence of execution time.

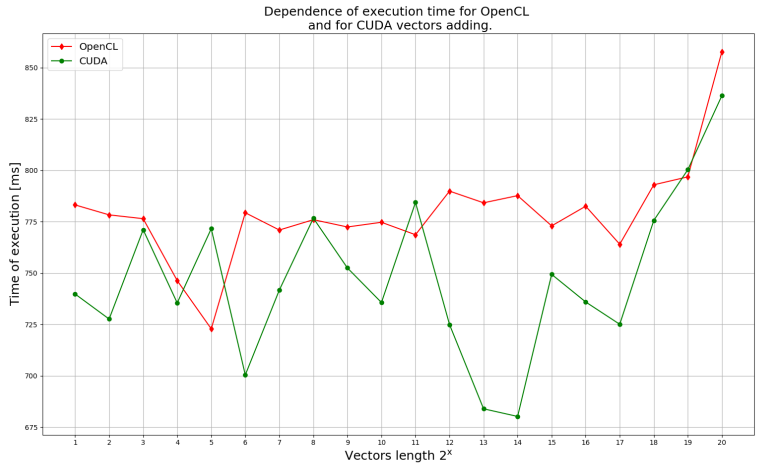

Figure 2: Dependence of execution time.

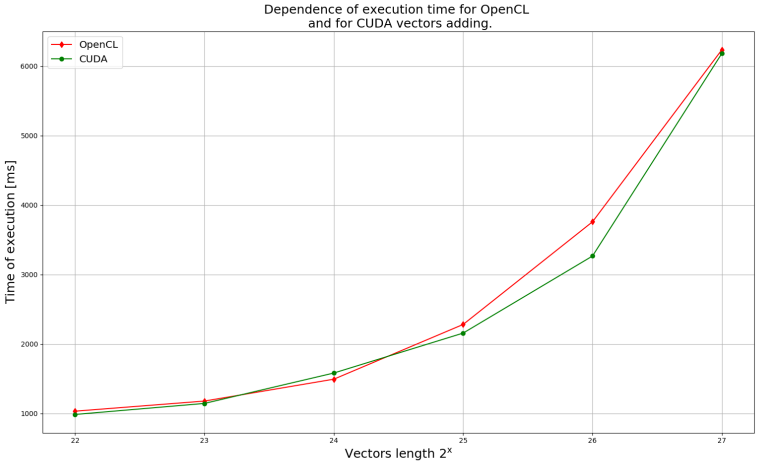

Figure 3: Dependence of execution time.

Based on data presented on pictures 1, 2 and 3 we can conclude that CUDA implementation is the best solution for vectors adding. Mostly of the time CUDA implementation is faster than OpenCL. However the differences are very small and consequently there are not a clear difference in performance using CUDA or OpenCL solution.

All collected data are also available to get from table 1:

Table 1: Table of all collected data

x: 2^x	OpenCL [ms]	CUDA [ms]
1	783.124	739.76
2	778.28	727.571
3	776.425	771.033
4	746.289	735.409
5	722.799	771.557
6	779.31	700.35
7	770.904	741.675
8	775.937	776.671
9	772.389	752.437
10	774.672	735.616
11	768.608	784.349
12	789.85	724.87
13	784.158	683.863
14	787.648	680.141
15	772.924	749.315
16	782.499	735.877
17	763.989	725.019
18	792.927	775.565
19	796.795	800.283
20	857.698	836.289
21	923.691	838.187
22	1033.22	985.969
23	1178.69	1144.72
24	1493.81	1582.42
25	2281.91	2157.64
26	3759.99	3267.12
27	6237.36	6184.5
28	11094.5	11249.2
29	-	21860.8
30	-	45317.1