

CS607: Contemporary Computing Platforms

Project 2

Submitted by:
Harshit Malik
2017CSB1078

Requirements:

NVCC Version: 10.1.243
CUDA Version: 10.1

How to run the program:

→ Running on Goolge Colab

- ◆ Open the python notebook 'google_colab_script.ipynb' in google colab and run the cell.
- ◆ Select and upload the sgmev.cu or ztmpv.cu file and uncomment the commands accordingly.

→ Running on Local Machine

- ◆ Execute the sgmev.cu by the following command

```
$ nvcc sgmev.cu -lcublas -lcurand -o output  
$ ./output
```

- ◆ Execute the ztmpv.cu by the following command

```
$ nvcc ztmpv.cu -lcublas -lcurand -o output  
$ ./output
```

Changing Function Paramters:

- Each file sgmev.cu and ztmpv.cu contains function parameters as macros at the starting of the file. They can be changed accordingly. They have been well commented to describe them.
- As the matrix dimensions can be very large, matrices and vectors have been filled initially randomly. To fill manual entry one has to change the init_vals() function accordingly.

CuBLAS Routines Used:

- SGMEV: Level 3 function SGEMM (<http://www.netlib.org/blas/sgemm.f>)
- ZTMPV: Level 2 function STPMV (<http://www.netlib.org/blas/stpmv.f>)

Results

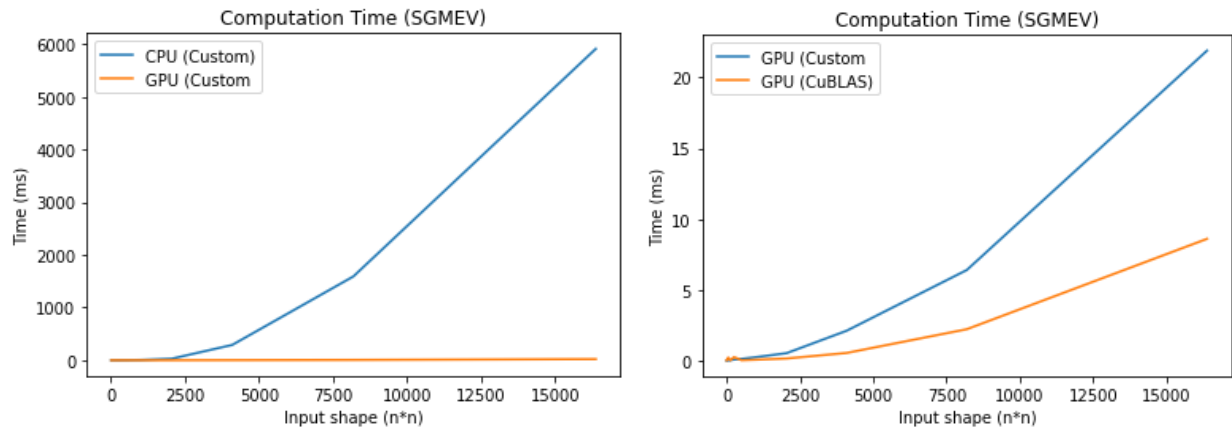
➔ Routine 1: SGMEV

$y := \alpha * op(A) * x + \beta * y$

Performance Comparison Table for SGMEV routine

Input Shape	CPU Time (in ms)	GPU Time (Custom) (in ms)	GPU Time (CuBLAS) (in ms)
4*4	0.002	0.030	0.043
8*8	0.002	0.031	0.042
16*16	0.003	0.031	0.044
32*32	0.006	0.034	0.186
64*64	0.016	0.041	0.241
128*128	0.057	0.059	0.042
256*256	0.328	0.103	0.277
512*512	1.643	0.150	0.056
1024*1024	6.442	0.277	0.093
2048*2048	28.883	0.554	0.178
4096*4096	290.499	2.136	0.568
8192*8192	1591.48	6.419	2.242
16384*16384	5912.3	21.896	8.618

Performance Graphs



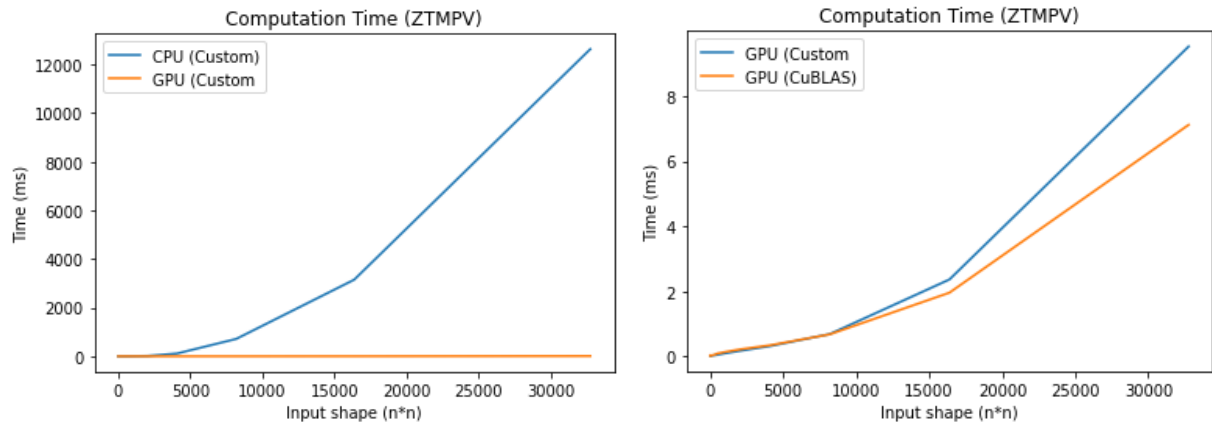
→ Routine 2: ZTMPV

$x := op(A) \cdot x$

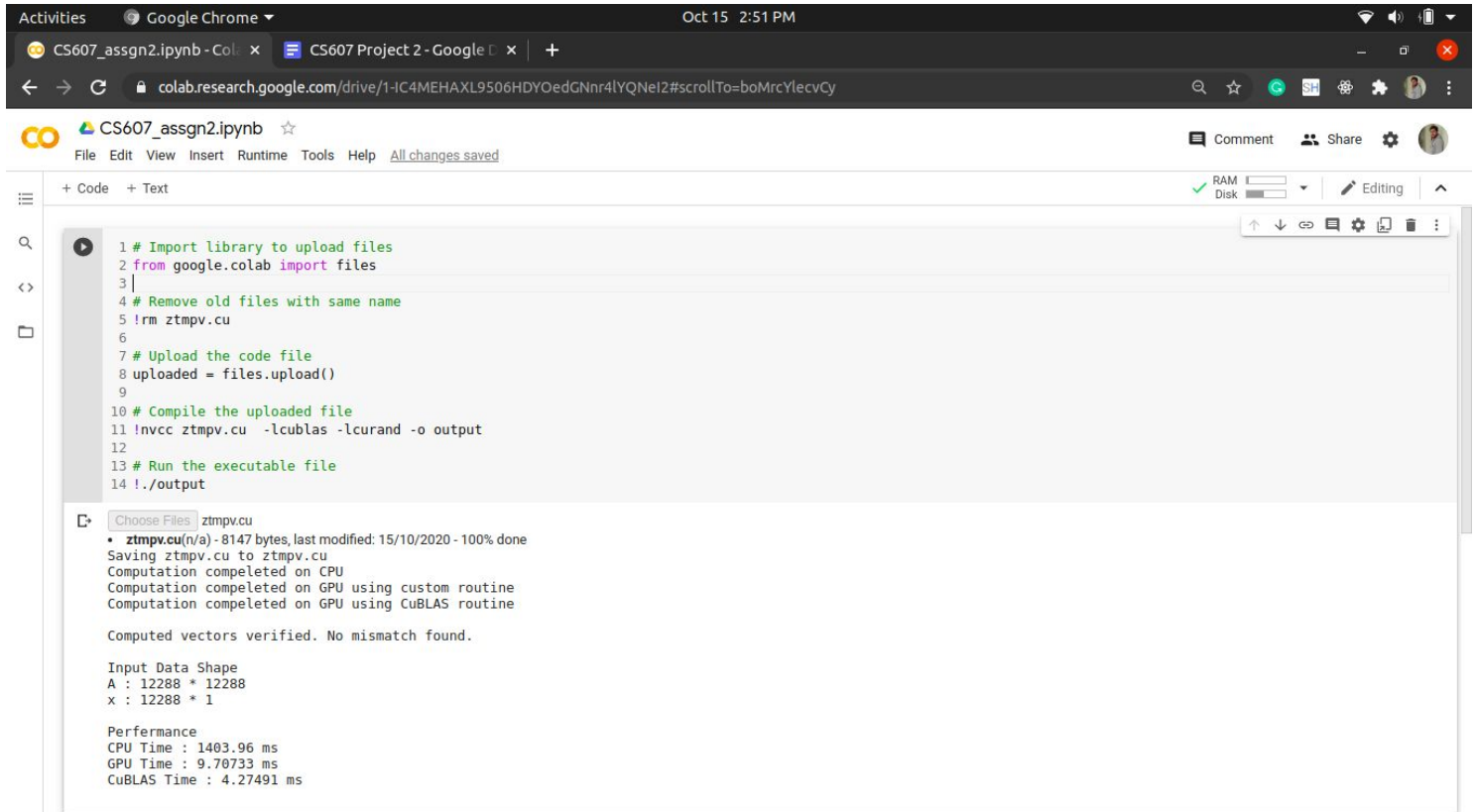
Performance Comparison Table for ZTMPV routine

Input Shape	CPU Time (in ms)	GPU Time (Custom) (in ms)	GPU Time (CuBLAS) (in ms)
4*4	0.002	0.022	0.028
8*8	0.002	0.022	0.027
16*16	0.003	0.032	0.040
32*32	0.006	0.033	0.028
64*64	0.010	0.024	0.032
128*128	0.032	0.029	0.036
256*256	0.139	0.035	0.047
512*512	0.681	0.052	0.094
1024*1024	4.595	0.097	0.138
2048*2048	13.825	0.173	0.219
4096*4096	114.993	0.318	0.344
8192*8192	713.134	0.688	0.680
16384*16384	3149.2	2.364	1.958
32768*32768	12628.6	9.520	7.114

Performance Graphs



Screenshots of Execution on Google Colab



This screenshot shows the execution of a Jupyter Notebook in Google Colab. The notebook is titled "CS607_assgn2.ipynb". The code in the cell performs the following steps:

- Import library to upload files
- from google.colab import files
-
- Remove old files with same name
- !rm ztmpv.cu
-
- Upload the code file
- uploaded = files.upload()
-
- Compile the uploaded file
- !nvcc ztmpv.cu -lcublas -lcurand -o output
-
- Run the executable file
- !./output

The output of the execution is as follows:

Choose Files: ztmpv.cu

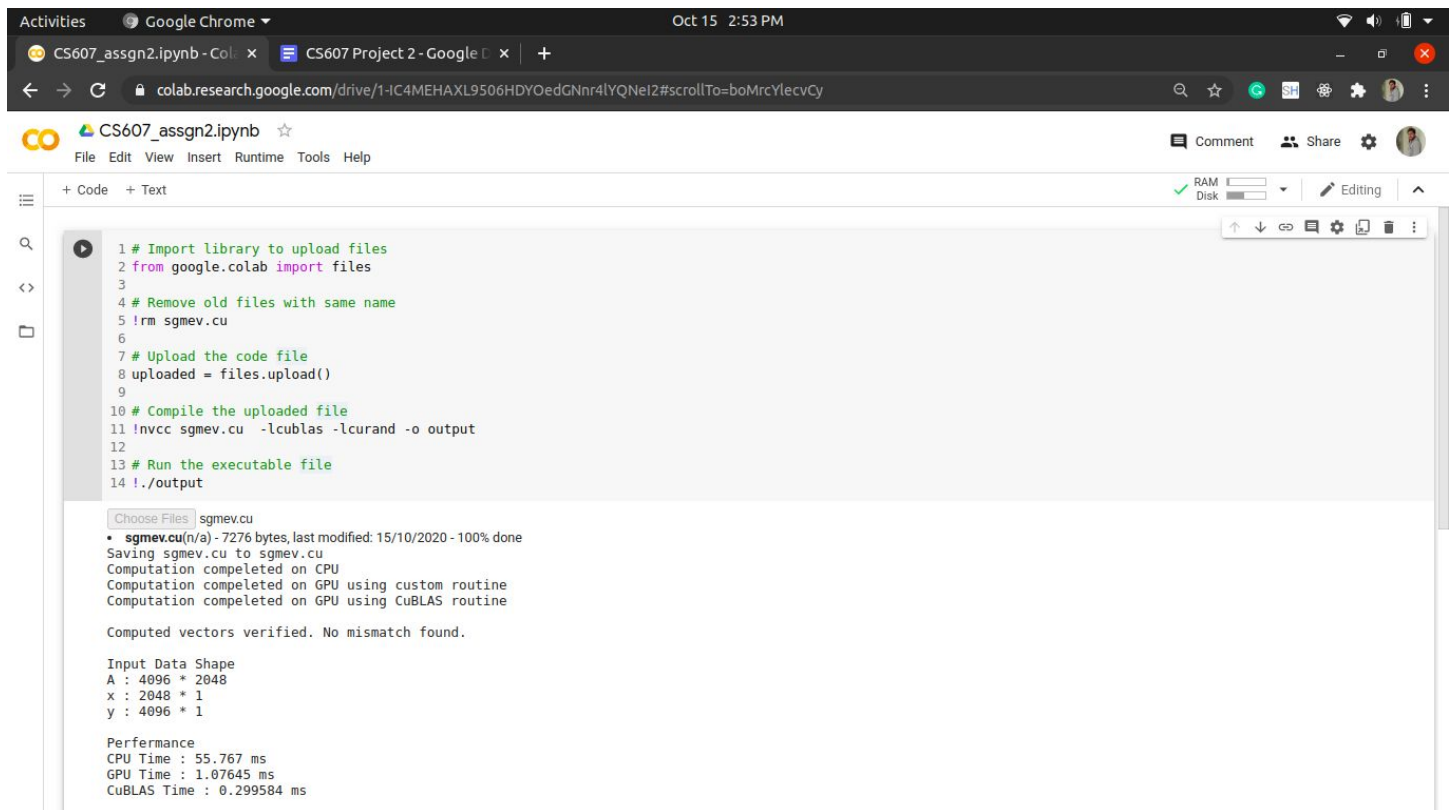
- ztmpv.cu(n/a) - 8147 bytes, last modified: 15/10/2020 - 100% done

Saving ztmpv.cu to ztmpv.cu
Computation completed on CPU
Computation completed on GPU using custom routine
Computation completed on GPU using CuBLAS routine

Computed vectors verified. No mismatch found.

Input Data Shape
A : 12288 * 12288
x : 12288 * 1

Performance
CPU Time : 1403.96 ms
GPU Time : 9.70733 ms
CuBLAS Time : 4.27491 ms



This screenshot shows the execution of a Jupyter Notebook in Google Colab. The notebook is titled "CS607_assgn2.ipynb". The code in the cell performs the following steps:

- Import library to upload files
- from google.colab import files
-
- Remove old files with same name
- !rm sgmev.cu
-
- Upload the code file
- uploaded = files.upload()
-
- Compile the uploaded file
- !nvcc sgmev.cu -lcublas -lcurand -o output
-
- Run the executable file
- !./output

The output of the execution is as follows:

Choose Files: sgmev.cu

- sgmev.cu(n/a) - 7276 bytes, last modified: 15/10/2020 - 100% done

Saving sgmev.cu to sgmev.cu
Computation completed on CPU
Computation completed on GPU using custom routine
Computation completed on GPU using CuBLAS routine

Computed vectors verified. No mismatch found.

Input Data Shape
A : 4096 * 2048
x : 2048 * 1
y : 4096 * 1

Performance
CPU Time : 55.767 ms
GPU Time : 1.07645 ms
CuBLAS Time : 0.299584 ms