ME 766: High Performance Scientific Computing

Assignment 03

Aaron John Sabu

1 Introduction

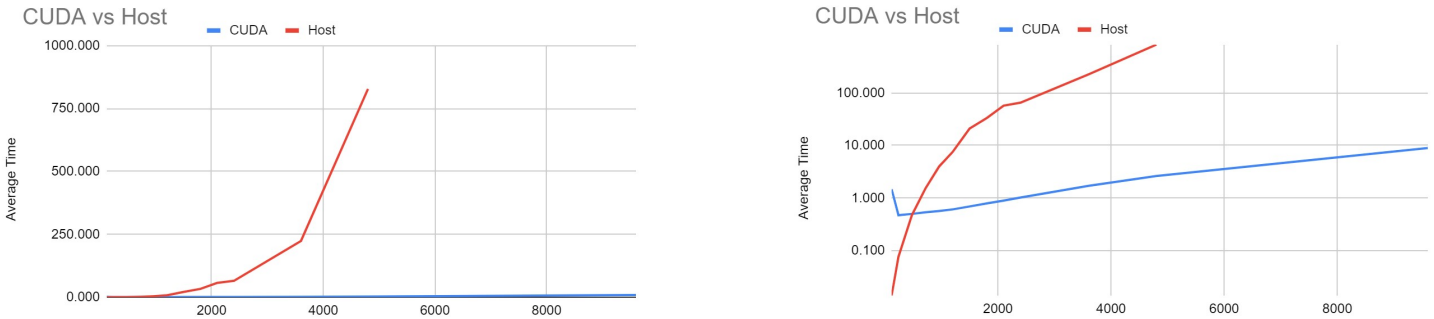
This assignment implements matrix multiplication of two $N \times N$ matrices after their random initialization. This implementation is performed both on the host (CPU) and on the device (GPU) using CUDA. The following sections provide insight into the work done for the same.

2 Timing Study Results

We run the codes over specific values of N for 10 times and consider the average time taken for each value of N. The CUDA code runs matrix multiplication on the GPU while the C code runs all operations on the CPU.

N	Host	CUDA	N	Host	CUDA
120	0.014	1.467	1800	33.073	0.788
240	0.076	0.470	2100	57.073	0.892
480	0.474	0.498	2400	65.073	1.020
720	1.532	0.535	3600	223.498	1.702
960	3.977	0.566	4800	827.672	2.620
1200	7.532	0.606	9600	TIME LIMIT	8.901
1500	21.001	0.690		EXCEEDED	

3 Comparison Plot for Timing Study



(a) Linear Scale

(b) Logarithmic Scale

Figure 1: Timing Study: CUDA vs Host

4 Observations

We observe that the time taken for the CUDA code to run on the device (GPU) is much smaller than that taken for the C code to run on the host (CPU) due to its parallelization. In particular, the host (CPU) takes hours to calculate the product of two 9600×9600 matrices while the device (GPU) performs the same operation in less than 10 seconds. This is a clear indication of the advantages of using GPUs for repetitive and monotonous operations.

A Individual Data Points

N	Host										CUDA									
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
120	1.644	1.909	1.280	1.294	1.294	1.221	1.212	2.057	1.247	1.510	0.017	0.017	0.007	0.008	0.018	0.016	0.017	0.014	0.008	0.017
240	0.490	0.484	0.486	0.483	0.445	0.476	0.455	0.466	0.451	0.459	0.085	0.085	0.083	0.067	0.085	0.080	0.084	0.056	0.048	0.088
480	0.519	0.483	0.526	0.482	0.477	0.498	0.474	0.517	0.515	0.489	0.482	0.484	0.487	0.475	0.491	0.464	0.491	0.438	0.465	0.461
720	0.530	0.510	0.531	0.524	0.496	0.529	0.509	0.549	0.532	0.639	1.470	1.463	1.664	1.407	1.561	1.537	1.746	1.486	1.534	1.455
960	0.561	0.561	0.552	0.576	0.548	0.555	0.655	0.544	0.555	0.550	3.428	4.013	4.151	4.069	4.067	3.999	4.267	3.992	4.003	3.781
1200	0.619	0.611	0.617	0.617	0.585	0.614	0.609	0.578	0.605	0.607	6.570	7.803	7.758	7.347	7.726	8.019	7.095	7.536	7.944	7.519
1500	0.679	0.703	0.682	0.715	0.673	0.677	0.713	0.676	0.682	0.703	16.491	21.433	21.586	22.588	19.241	22.363	21.887	21.946	22.688	19.782
1800	0.765	0.813	0.769	0.808	0.780	0.785	0.800	0.785	0.808	0.769	28.641	33.719	33.885	34.766	31.511	34.267	33.939	33.877	34.550	31.579
2100	0.868	0.963	0.884	0.868	0.889	0.889	0.881	0.867	0.891	0.916	53.977	58.897	58.877	61.064	41.614	60.773	60.566	60.139	62.531	52.287
2400	0.997	1.044	1.021	1.021	0.993	1.007	1.029	1.037	1.008	1.039	63.014	66.729	66.762	66.948	63.762	65.618	66.566	65.597	65.901	59.830
3600	1.683	1.749	1.660	1.678	1.788	1.664	1.688	1.696	1.732	1.683	213.153	225.017	223.318	226.980	223.100	225.955	222.534	224.995	226.261	223.667
4800	2.565	2.644	2.614	2.643	2.598	2.631	2.620	2.622	2.620	2.644	765.930	816.761	841.448	851.790	833.181	824.568	815.166	844.346	852.179	831.353
9600	8.783	8.867	8.785	8.972	9.054	8.859	8.921	8.942	8.855	8.975	TLE	CNC	CNC	CNC	CNC	CNC	CNC	CNC	CNC	CNC

(TLE = Time Limit Exceeded, CNC = Cancelled anticipating TLE)

B Configurational Detail

The host code was run on the supercomputer (PARAM Sanganak) for which the author is very highly grateful to the instructor and C-DAC. The CUDA part was performed on the laptop of the author which has the following characteristics:

- **CPU**
  - Specification: Intel Core i5-10300H CPU @ 2.50GHz
  - Cores: 4
  - Threads: 8
- **RAM**: 8192 MB
- **Operating System** : Windows 10 Home Single Language 64-bit
- **Graphics Processing Unit** : NVIDIA GeForce GTX 1650 Ti