CS 531 HW 4

Hritvik

November 2025

1. Introduction

This report analyzes the performance of GPU-based Sparse Matrix-Vector Multiplication (SpMV) using two formats: Compressed Sparse Row (CSR) and ELLPACK/ITPACK (ELL). The goal is to evaluate how GPU performance changes under different CUDA thread block sizes using the `cant.mtx` matrix. The CPU implementation serves as a correctness reference.

2. Experimental Setup

All runs were executed on a Talapas GPU node using:

- 1 NVIDIA GPU
- CUDA 11.8
- OMP with 28 CPU threads (CPU code only)

The primary metric recorded is kernel execution time per iteration.

3. Results

Table 1 summarizes the GPU CSR and ELL SpMV execution times for different thread block sizes.

Block Size	**CSR Time (s)**	**ELL Time (s)**	**2-Norm Error**
32	0.00040677	0.00048979	1.77×10^{-10}
64	0.00065153	0.00049024	1.77×10^{-10}
128	0.00127685	0.00048880	1.77×10^{-10}
256	0.00019628	0.00008211	1.77×10^{-10}

Table 1: GPU SpMV performance across block sizes.

4. Analysis

CSR performance did not scale monotonically with block size. Performance degraded at 64 and 128 threads but improved significantly at 256, suggesting better latency hiding and memory aggregation at larger block sizes.

ELL performance was stable for block sizes 32–128 due to its padded, uniform structure. At 256 threads, ELL achieved a large speedup, becoming the fastest configuration tested. This behavior is consistent with ELL's regular memory access pattern, which benefits from larger blocks.

Across all configurations, ELL outperformed CSR at 256 threads by a large margin.

5. Conclusion

Both CSR and ELL GPU implementations produced correct results. CSR exhibited variable performance across block sizes, while ELL showed stable behavior with a dramatic improvement at 256 threads. Overall, ELL with a block size of 256 was the most efficient configuration for the `cant.mtx` matrix.