Cuda Optimized SpMM Milestone Report

Mingxin Li & Zephyr Zhao

Current Progress

We have finished several goals as mentioned in our proposal. We finished setting up the environment, which includes selecting test cases from SuiteSparse Matrix Collection, format conversion tools, and validation scripts. We also finished implementing CPU versions of SpMM for two most commonly used storage formats, and the basic CUDA kernels for those formats as well. We are in the process of gathering more matrices for testing and evaluating the performance of our implementation when handling matrices of different characteristics. We are also in the process of optimizing CUDA performance by profiling and fine-tuning.

Revised Plan

After discussing our project scope with the professor, we realized that our stretch goal might be unrealistic for this class, as optimizing SpMM using advanced hardware like the A100 necessitates working with assembly-like languages, which are highly challenging to implement. Since we have the resources, we decide that we will try to learn the framework and put in some effort and see if we can finish a simple working implementation if time permits. To compensate, we will explore more storage formats such as block compressed row(BSR) formats, and perform benchmark testing and analysis.

Our revised weekly plans from 12/2 - 12/15 areas follows:

Date	Plan
12/2 - 12/5	Finish additional format implementation
12/5 - 12/8	Collecting more test cases; Benchmark against cuSparse
12/8 - 12/13	Analysis for the poster; Explore advanced hardware features if time permits
12/13 - 12/15	Final Report; Organize code base

Revised Current Goals and Deliverables

1. 50% goal:
    - Done: Implement CPU versions as baseline and verifier serving our GPU version
    - Done: ~~Randomly generate sparse matrix for correctness test.~~ Selected matrices from SuiteSparse Matrix Collection
    - Done: Implement basic CUDA kernels without any performance requirements
    - In Progress: ~~Analyze different storage formats to choose the best one~~ Implementing more formats to choose the best
2. 75% goal:
    - Done: Integrate cuSparse functionality
    - In Progress: Use advanced CUDA techniques like shared memory to reduce memory traffic, mitigate warp divergence, avoid bank conflicts and improve throughput
    - In Progress: Implement more storage formats and analyze (dis)advantages of different formats
    - In Progress: Thoroughly benchmark our kernel implementation with cuSparse
3. 100% goal:
    - In Progress: Fine-tune (together with advanced CUDA techniques in 75% goal)
    - Not Started: Conduct comprehensive experiments and evaluate our implementation from different perspectives
    - Not Started: Achieve near 50% throughput of cuSparse.
4. 125% goal
    - Collect real-word LLM weight matrices for final evaluation
    - Explore optimization for advanced hardware such as A100

Poster Session Plan

We plan to display our comparison of different formats and benchmarking results in addition to the problem statement and challenges of SpMM. We plan to provide graphs of speedup like we did in project 3 and 4 to help our audiences easily understand the performance of our kernels. We plan to also use short bullet points to explain the reasons for gains or losses in performance. Analysis based on tests and profiling results will also be presented in detail.

Concerns

As discussed above, we have concerns about optimization for advanced hardware because of implementation challenges. We are also concerned about achieving our goal of 50% performance of cuSparse because we found out there are many advanced CUDA features to explore and the workload is a little bit heavy for only 2 weeks of time. We will put more effort into the project and explore more techniques to improve kernel performance.