CMOR 421/521 Assignment 3: Using MPI to implement Cannon's algorithm and SUMMA algorithm

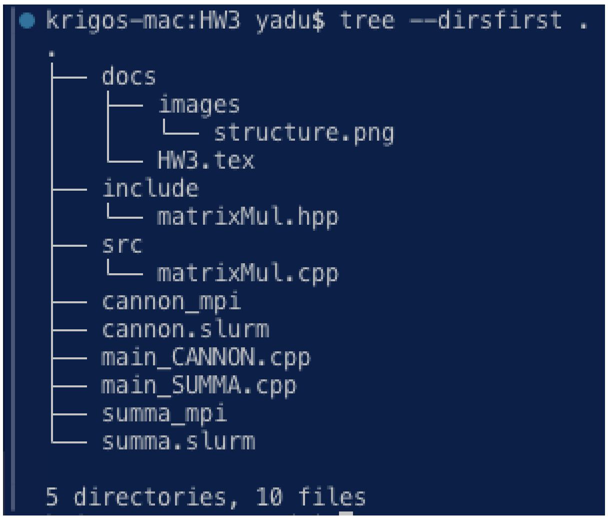
Yuhao Liu

April 18, 2025

Contents

1 Directory Structure

Below is my file organization for this assignment. My final zip file follows this structure (`docs/` for LaTeX, `src/` for source files, and `include/` for header files):



```
krigos-mac:HW3 yadu$ tree --dirsfirst .
.
├── docs
│   ├── images
│   │   └── structure.png
│   └── HW3.tex
├── include
│   └── matrixMul.hpp
├── src
│   └── matrixMul.cpp
├── cannon_mpi
├── cannon.slurm
├── main_CANNON.cpp
├── main_SUMMA.cpp
├── summa_mpi
└── summa.slurm

5 directories, 10 files
```

Figure 1: structure

- The `include` folder and `src` folder have `matrixMul.hpp` and `matrixMul.cpp`. In `matrixMul.cpp`, there are 2 help function
  - `void testMul(const int N, double* serialC, double* mpiC)`

    This function help to test that the result from both algorithms are equal to the matrix computed by serial matrix multiplication.
  - `void serialMatMult(const int N, double* C, const double*`

    This function is used to compute the matrix in serial version.
- The `main_CANNON.cpp` and `main_SUMMA.cpp` contain the Cannon's algorithm and SUMMA algorithm respectively.
- The `cannon_mpi` and `summa_mpi` are executable files. Both of them already complied by `-O3` optimization flag.
- The `cannon.slurm` and `summa.slurm` are sbatch scripts.

2 How to Build and Run the Code (In NOTXs)

- **Build and run**
  - For each algorithm you can using

    ```
    sbatch summa.slurm
    sbatch cannon.slurm
    ```

    to run these 2 different algorithm with 2×2, 3×3, 4×4 grids with $N = 512, 768, 1024$ and $k = 64$. All these parameters can be found in `*.slurm` file

- **The results**
  - SUMMA algorithm:

    ```
    Job running on nodes: bb2u12c1,bb2u14c1
    ==================================================
    === Running SUMMA on 4 procs (2×2 grid) ===
    Total error = 1.54811e-08
    SUMMA multiplication is CORRECT.
    === Running SUMMA on 9 procs (3×3 grid) ===
    Total error = 7.07518e-08
    SUMMA multiplication is CORRECT.
    === Running SUMMA on 16 procs (4×4 grid) ===
    Total error = 1.72552e-07
    SUMMA multiplication is CORRECT.
    All runs complete.
    ```

  - Cannon's algorithm:

    ```
    Job running on nodes: bb2u12c1,bb2u14c1
    ==================================================
    === Running CANNON on 4 procs (2×2 grid) ===
    Total error = 1.1352e-08
    SUMMA multiplication is CORRECT.
    === Running CANNON on 9 procs (3×3 grid) ===
    Total error = 6.90912e-08
    SUMMA multiplication is CORRECT.
    === Running CANNON on 16 procs (4×4 grid) ===
    Total error = 1.62022e-07
    SUMMA multiplication is CORRECT.
    All runs complete.
    ```