

# Project Checkpoint

Zhao Chen [zhaoc2@andrew.cmu.edu](mailto:zhaoc2@andrew.cmu.edu)

Jianlin Du [jianlind@andrew.cmu.edu](mailto:jianlind@andrew.cmu.edu)

## Web Page URL:

<https://deltadu.github.io/Parallel-Floyds/>

- Make sure your project schedule on your main project page is up to date with work completed so far, and well as with a revised plan of work for the coming weeks. As by this time you should have a good understanding of what is required to complete your project, I want to see a very detailed schedule for the coming weeks. I suggest breaking time down into half-week increments. Each increment should have at least one task, and for each task put a person's name on it.

Week of 11/18 first half:

Come up with MPI version of the algorithm and discuss -- Together

Week of 11/18 second half:

Implement MPI version of the algorithm -- Together

Week of 11/25 first half:

Compare MPI version and OpenMP version performance -- Jianlin

Explore different partition strategy of parallelization in OpenMP and MPI -- Zhao

Week of 11/25 second half:

Try 'nice-to-have': combine the OpenMP and MPI to achieve a faster performance on multi nodes multi cores cluster, and benchmark it -- Together

Week of 12/2 first half:

Design the demo, define interactive input and output -- Together

Week of 12/2 second half:

Write final report and make poster -- Together

- One to two paragraphs, summarize the work that you have completed so far. (This should be easy if you have been maintaining this information on your project page.)

We have finished the sequential version of algorithm and shared-memory parallel version by OpenMP. We explore some for-loop to parallelize the computation. We use adjacency matrix to represent the graph, partition the matrix into blocks by threads. Threads will share certain information to update the matrix.

- Describe how you are doing with respect to the goals and deliverables stated in your proposal. Do you still believe you will be able to produce all your deliverables? If not, why? What about the "nice to have's"? In your checkpoint writeup we want a new list of goals that you plan to hit for the poster session.

We are on a good track and completed our goals as scheduled. We believe we can produce all our deliverable in the end, although the tasks in the latter half of the plan are more complicated.

The schedule for the next few weeks are almost the same as before, so the goals include implementing MPI version, comparing MPI version with OpenMP version, Exploring different partition strategies and tuning hyperparameters, making interactive interface, making poster and writing the report. We leave one week to try an OpenMP and MPI combined version.

- What do you plan to show at the poster session? Will it be a demo? Will it be a graph?

For starters, there will be a graph of execution speed of our different parallel implementation vs the size of the graph, and it will be compared side by side with the speed graph of baseline sequential algorithm.

Furthermore, we plan to design a demo in which the viewer could input a graph and the algorithm will show the shortest paths between nodes so that the viewer could check the correctness of our algorithm.

- Do you have preliminary results at this time? If so, it would be great to included them in your checkpoint write-up.

The current performance of sequential and parallel version on graph with 1000 nodes is similar, about 5.5 s. We are working on to improve the OpenMP performance.

(Time calculated by timing.h used in previous assignments)

- List the issues that concern you the most. Are there any remaining unknowns (things you simply don't know how to solve, or resource you don't know how to get) or is it just a

matter of coding and doing the work? If you do not wish to put this information on a public web site you are welcome to email the staff directly.

We do not know how to combine the OpenMP and MPI on multi nodes multi cores cluster yet. We have not implemented such program in the previous assignment and we are not sure about what cluster to use for such program.