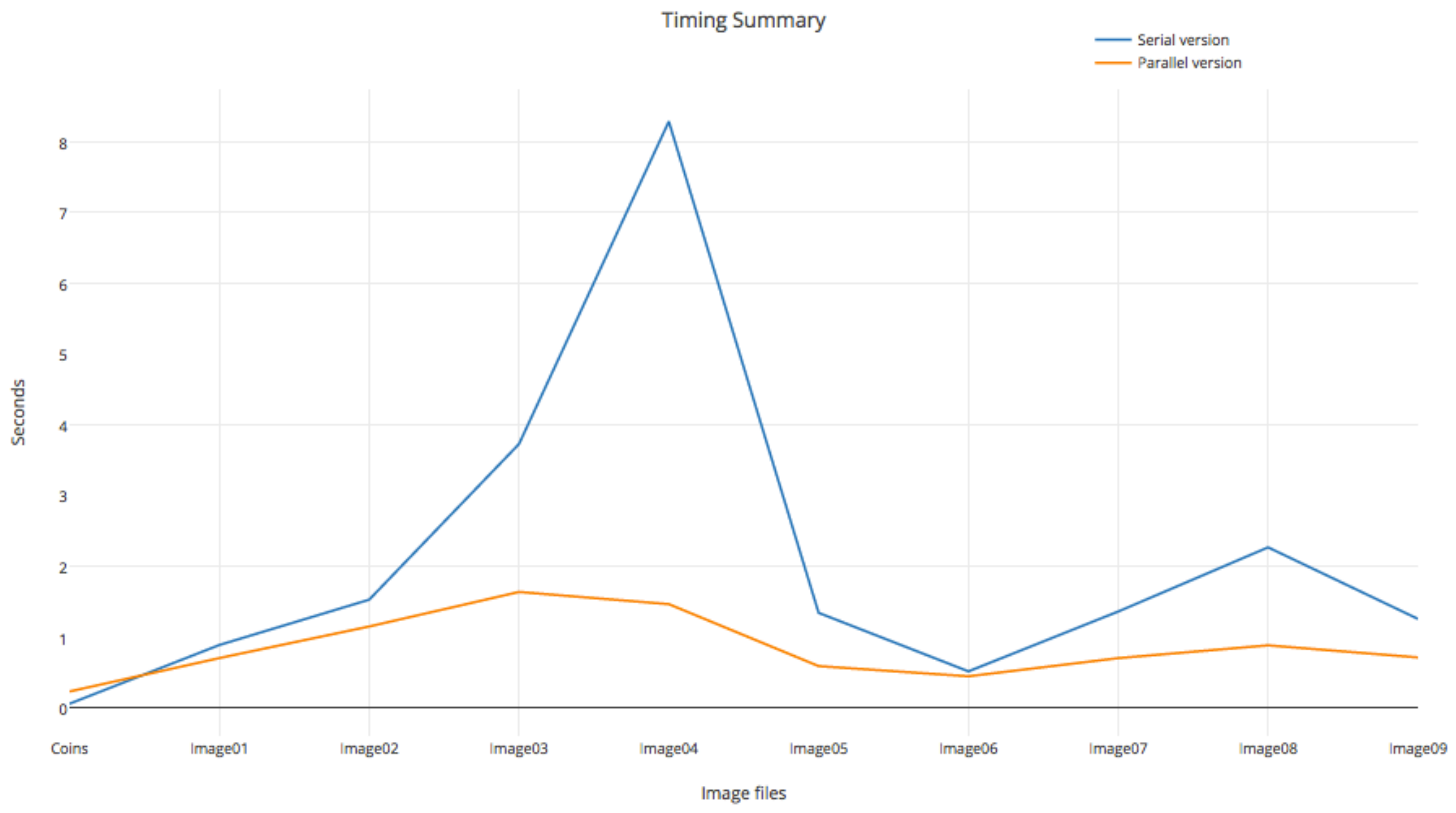
CSE 5441
Programming Assignment 5 – MPI

Section: 2:20 P.M.

Rajarshi Biswas (biswas.91@osu.edu)

Table of Contents

Timing Summary of Serial and Parallel (MPI + OpenMP) version:



- All the images converged with same threshold value for both the serial and parallel version

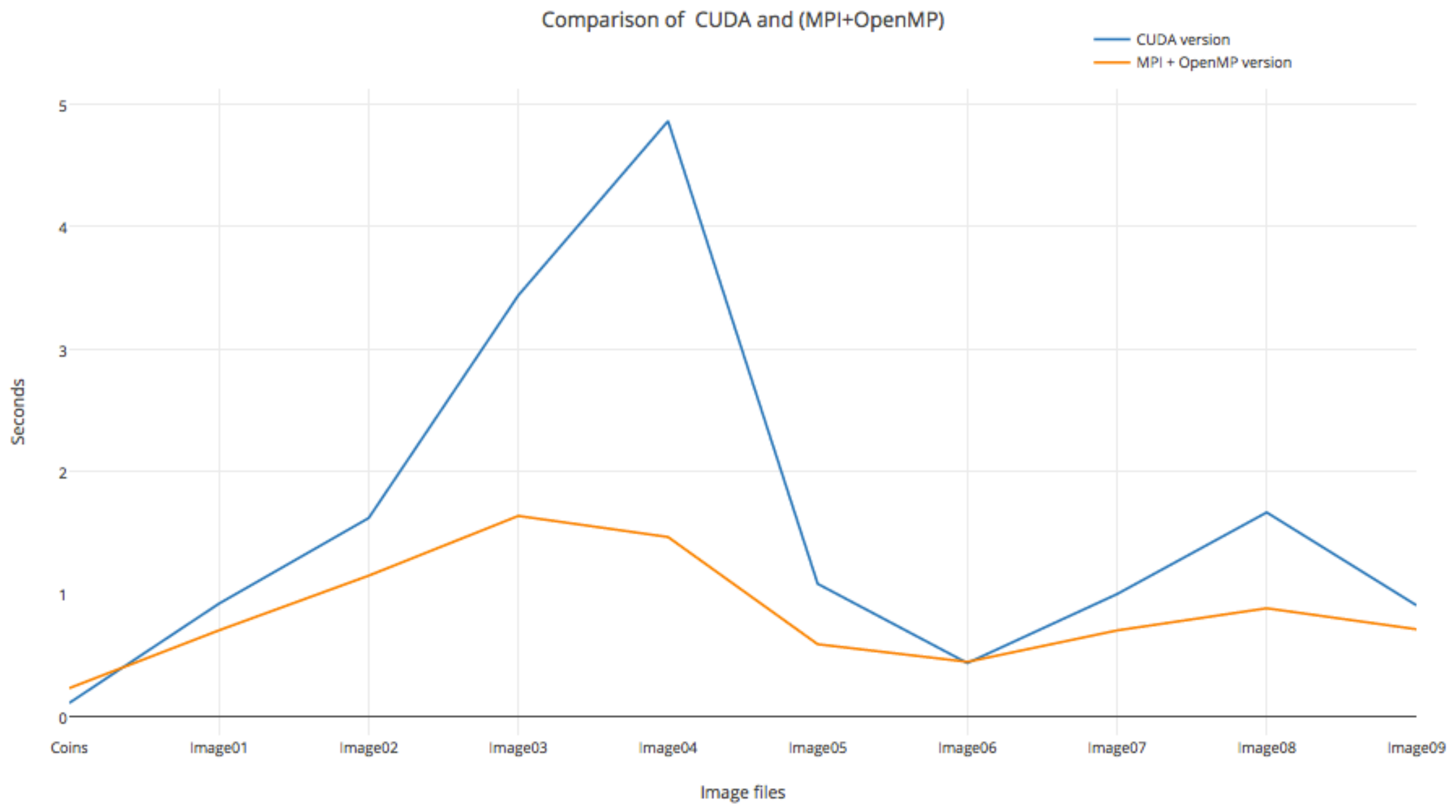
Performance:

- As expected the parallel version performs better than the serial version. Although, the serial version performed better for coins.bmp (as this image size is small, the parallel version could not perform better than the serial version due to the overhead of thread creation and communication).

Workload distribution:

- Every image is distributed (MPI) equally to 12 workers. All the workers get same number of rows to operate on. Only the last worker gets the extra rows if there are any. In addition, each worker node creates 32 threads (OpenMP) to operate on its portion of the data.

Comparison of CUDA and (MPI + OpenMP):



- As per my observation MPI + OpenMP performed slightly better than CUDA.

Reduction and Synchronization:

- All the worker (MPI) nodes are synchronized using MPI_Barrier.
- Each worker operates on its portion of data using threads created by OpenMP. They are synchronized using #pragma omp barrier. In addition, the black cell count on each thread are reduced (summation) using #pragma omp for reduction(+:black_cell_count).
- In addition, all the MPI workers communicates the value of their local black_cell_count using MPI_Allreduce.
- Furthermore, all the MPI workers send their work to the Master using MPI_Recv/ MPI_Send.

Surprises:

- I expected CUDA would perform better than MPI + OpneMP, but according to my observation MPI + OpenMP version performed well, although the margin is very small.