Project 3 - All-Pairs Shortest Paths

Team 1 – Bob Chen (kc853), Nimit Sohoni (nss66), Xiangyu Zhang (xz388)

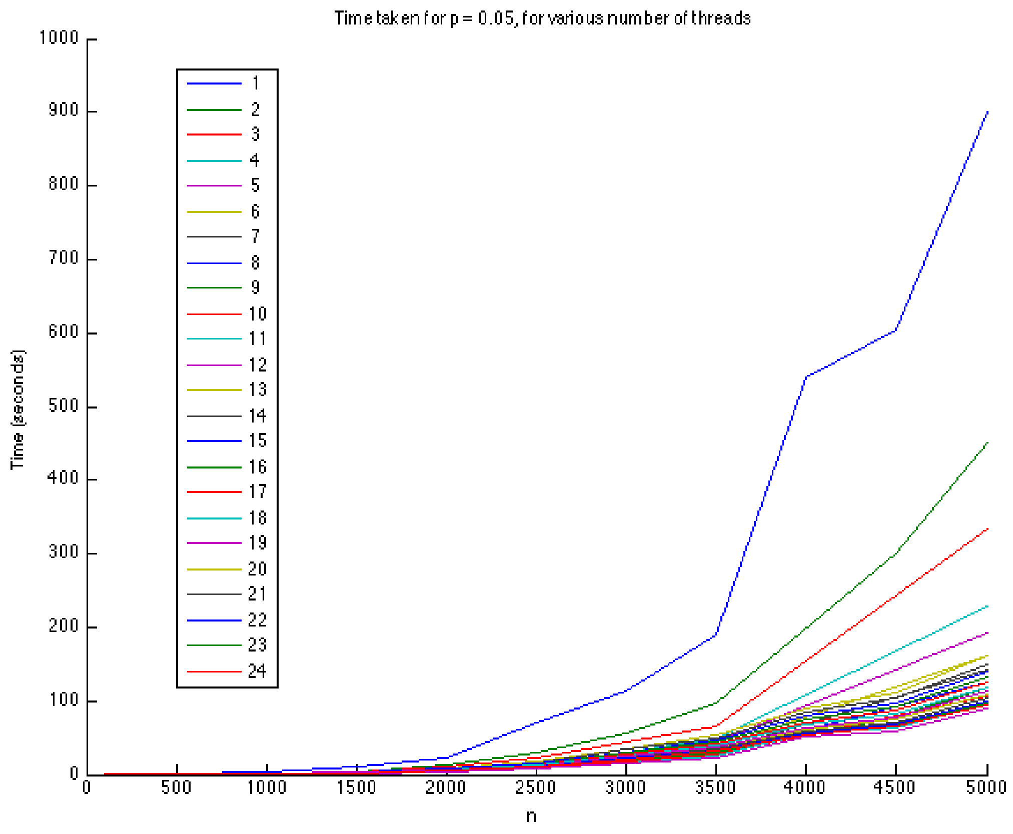
NOTE: This is an add-on to the rest of the report for Team 1.

1 MPI

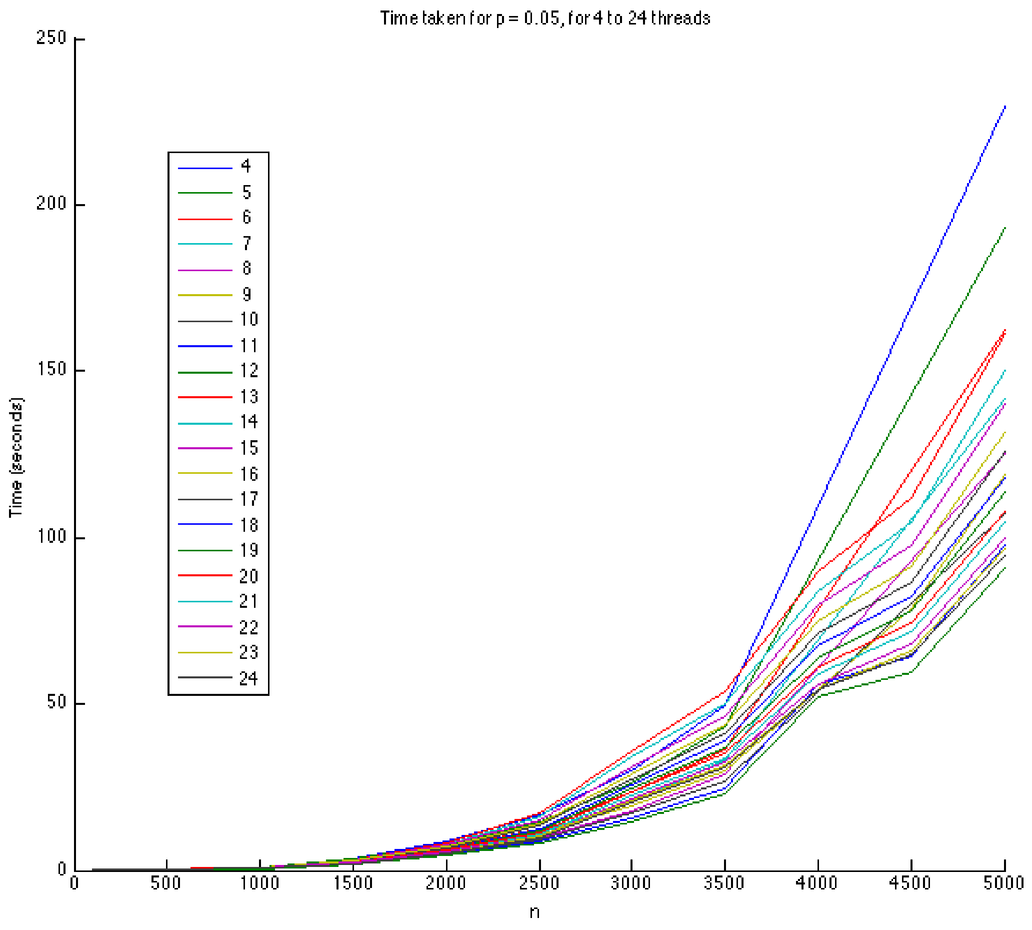
We also began to work on rewriting the code in an MPI fashion. We eventually decided to just focus on parallelizing the repeated squaring step with MPI, as the remainder of the steps run only once (gen_graph) and/or are much less costly. Each thread is assigned a block of columns to handle. In the square function, the thread multiplies its assigned columns by the entire L matrix on the left and, checks if the result is unequal to the result from the previous iteration, as in the original code. If any of the entries are unequal to the previous iteration, done is set to false. At each loop iteration, the results of 'done' from each thread are ANDed together. The blocks from each thread are gathered back into the new L matrix. We used the MPI_Allgatherv function because it is possible that n is not evenly divisible by the number of threads, in which case the leftover columns are assigned to the last thread.

The code for the MPI version is in `https://github.com/nimz/path/blob/master/path.c`

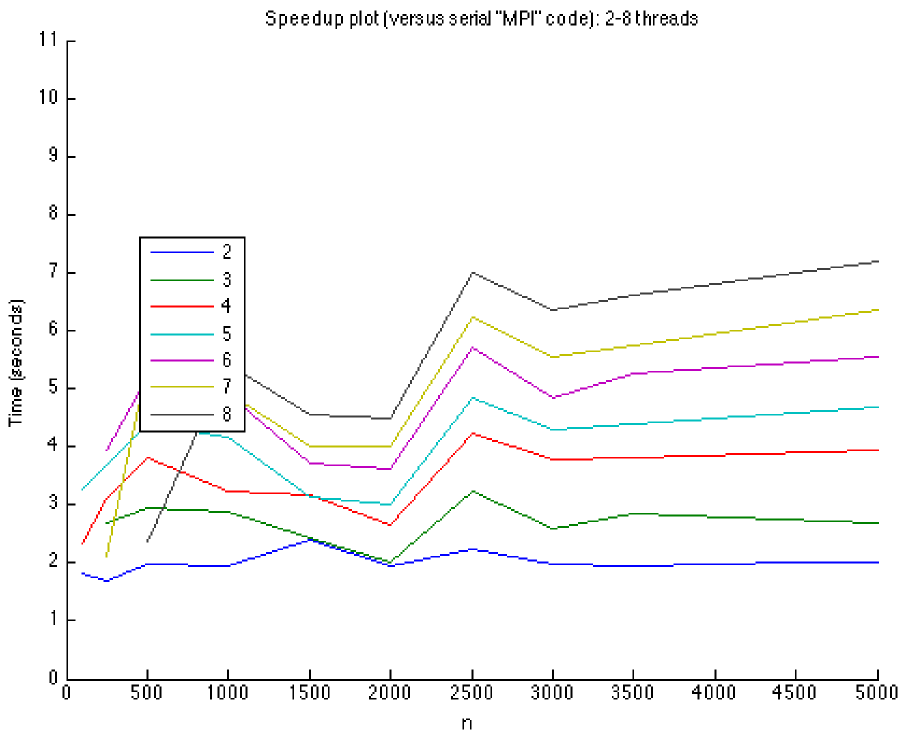
Timing results for varying number of threads:

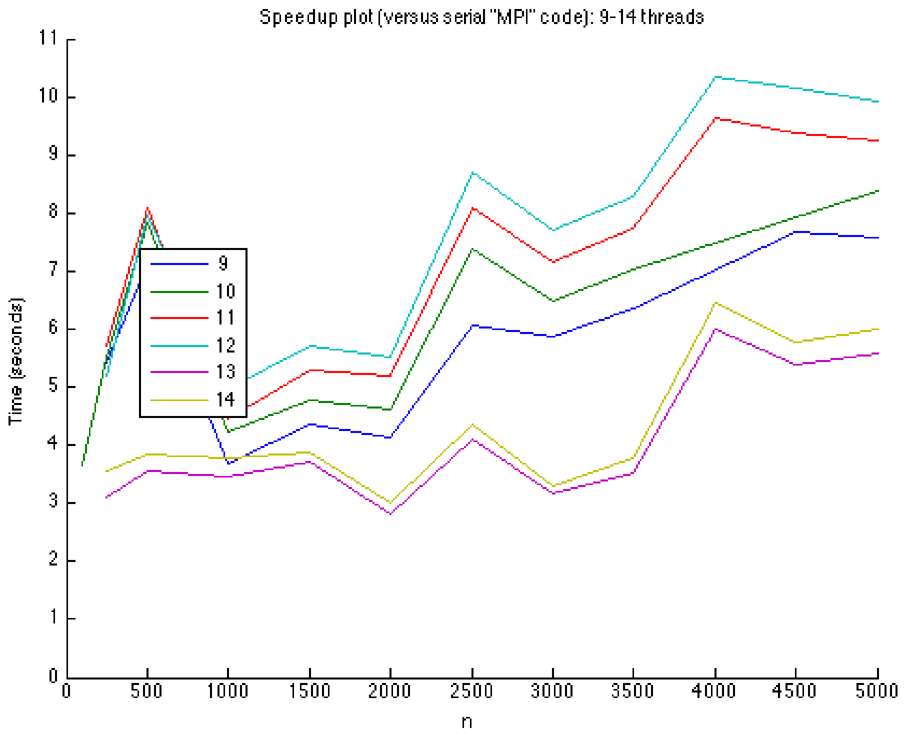


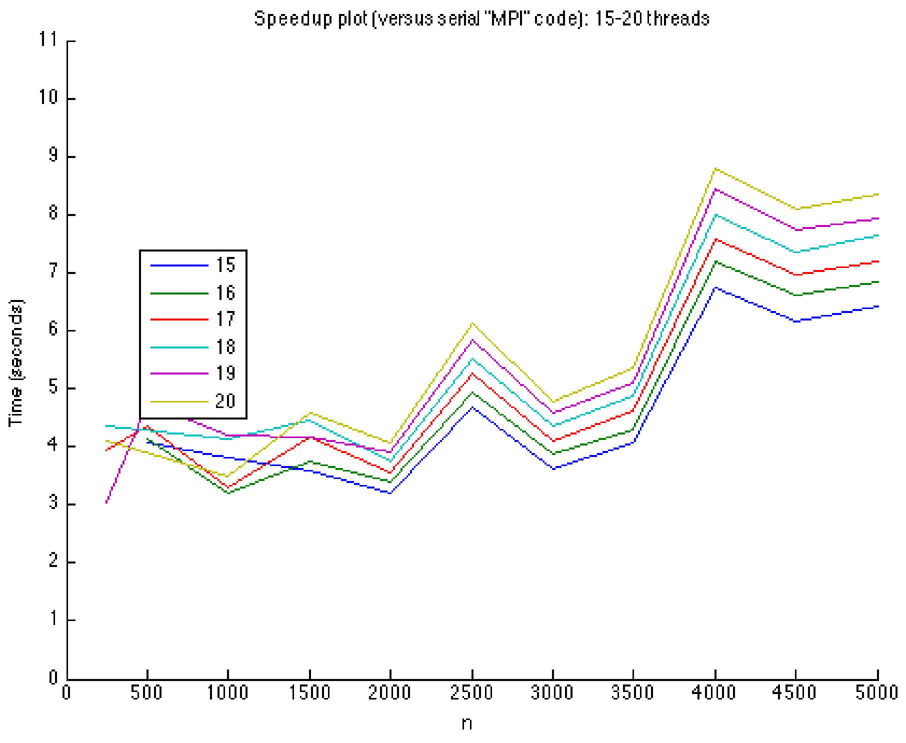
Timing results showing only with 4 to 24 threads for better visibility:

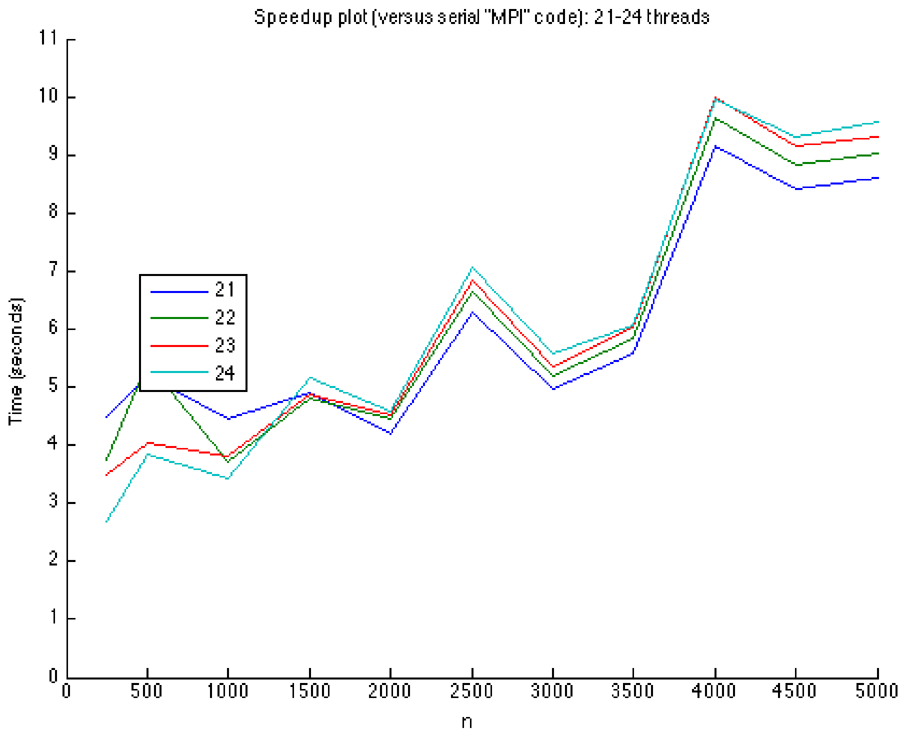


Speedup plots (split into several for readability):


Speedup plot (versus serial "MPI" code): 2-8 threads
Time (seconds)
n
2
3
4
5
6
7
8


Speedup plot (versus serial "MPI" code): 9-14 threads
Time (seconds)
n
9
10
11
12
13
14





Unsurprisingly, as we increase from 1 to 12 threads the speedup keeps increasing. However, once we go from 12 to 13 nodes we see a significant drop in speed. This is actually also unsurprising, because we have gone from 1 to 2 different chips (we have 12 cores / threads per chip), so there is communication overhead. After this, however, speedup continues as we increase the number of threads, as expected. With all 24 threads we get around 5x-10x speedup depending on the size of the problem.