SIT315 M3.S2P: Distributed Computing - MPI

Codey Funston | S222250824

No Group

Activity 1 – Distributed Vector Addition – MPI

With my implementations of vector addition programs running sequentially, with pthreads, with OpenMP, and with MPI, results were not as initially expected. Since I have an ARM-based machine I was using Docker containers as nodes in the MPI version and this was performing slower than all other three implementations. When I instead ran it directly on my machine and set the number of nodes to the number of cores I have (10) it ran much faster and was able to compete with the other implementations. Also, since we were required to use MPI\_Scatter() and MPI\_Gather(), the random filling of the two vectors was done sequentially by the master node. Previously, my pthreads and OpenMP programs did filling in parallel and so for accurate comparison I changed them to do that part sequentially. This also resulted in a significant drop in improvement against the sequential version which implies that the filling operation what benefits most from parallelisation.

