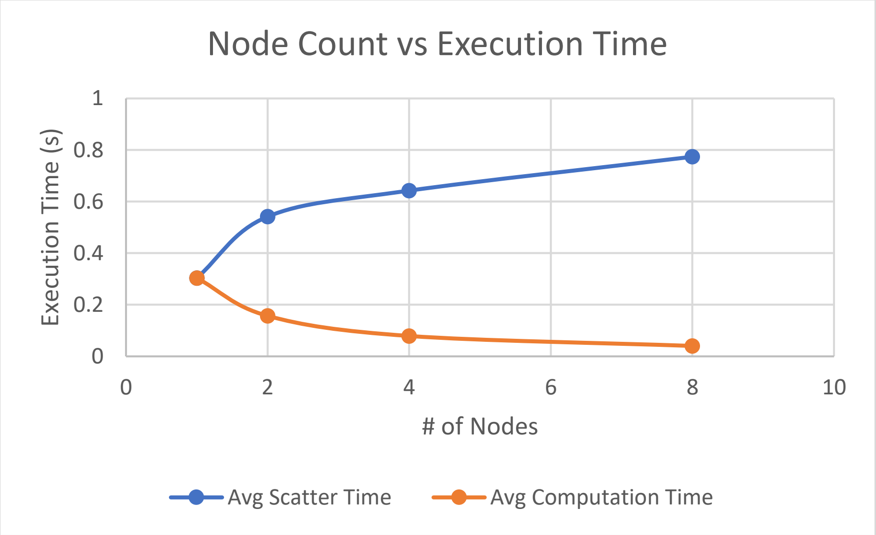
Michael Rivnak

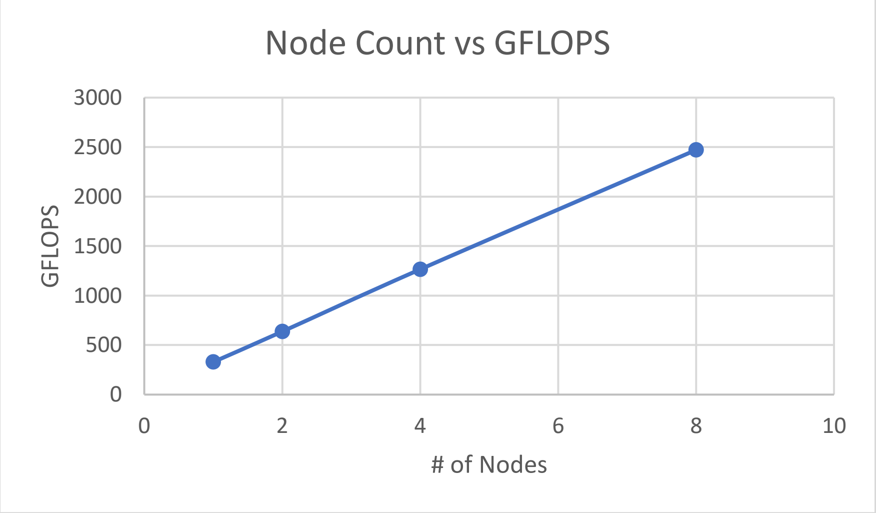
COMP 3450

Leonidas Deligiannidis

Assignment LA2

The maximum dimension of a matrix in the program is limited by the integer limit in the for loops that process the matrix. What I found to be happening with my program is if I increased the size too much the for loop iterator would overflow to a negative number and cause a segmentation fault. This could have been increased by changing those iterators to a unsigned or long type, but I felt that the data collected with the sizes I tested was representative of the program performance.





The data from this program showed to be very interesting, I was surprised with how well the performance seemed to scale. The calculation time was cut in half with each doubling of nodes, which makes sense, but I did not expect to see it this clearly in practice. This is also shown more impactfully on the Giga Floating Point Operations per Second (GFLOPS) graphs where it looks like a very linear increase in performance. What was also interesting is how quickly the scatter communication time rose with the increase in nodes. Looking at the graph would lead me to believe that there is a clear point of diminishing returns where the time saved through parallelization is less that the time lost through scattering data.