***Chart, line chart

Description automatically generatedCapi Graph***

According to Amdahls Law, the parallel program in Capi scales pretty good because as the number of threads increase, the speed-up increases as well (until the program uses over 64 threads, then it starts to almost even out). However, it is not quite getting to the 2x speed-up when increasing the threads like it should. Capi was the best system out of all three of the systems I ran the parallel program on due to the fact that it has so many cores (5 cores per socket and there are 4 sockets, so about 20 cores) and in each core there are 8 threads meaning there are about 160 threads in the Capi system. This allows the program to have plenty of space to run simultaneously among the many different threads which provides for more speed-up as you increase the number of threads used in the parallel program. There was also not as much running in the background as the other systems which allowed for faster times completing the program.

***Chart, line chart

Description automatically generatedZeus Graph***

According to Amdahls Law, the parallel program in Zeus scaled okay because for the most part the program got around a speedup of 3 which is similar to what Amdahls Law would have predicted. After 4 threads, the graph levels out meaning that the program would stay about the same time even if you increased the number of threads. This is because the Zeus system only has 4 cores and only one thread per core (4 in total), therefore the program would not be able to run the parallel program effectively on over 4 threads (causing the times to stay relatively the same as the thread count increased). The Zeus system was not as effective in parallel programming as the Capi program due to the difference in the number of cores and threads, however, I do believe that it was better than the Windows system, as well as more consistent.

***Chart, line chart

Description automatically generatedWindows Graph***

According to Amdahls Law, the parallel program in Windows did not scale very good because the graph is not super predictable and the points are all over the place, for example, the graph has a significant valley at the 32 thread mark. It seems as though the program would have done the same if not better and may have been faster if it was not a parallel program. This may be due to how many other programs were working in the background, which take up other parts of the CPU and the cores, and therefore other threads, making it not as fast as it should be according to Amdahls Law. My Windows system only has 4 cores in total, so a significantly less amount than the Capi system and likely a significantly less amount of threads to be working with, which is why the program ran in Capi had a lot more speed-up and was overall more efficient.