Adam Wright

wrighada@oregonstate.edu

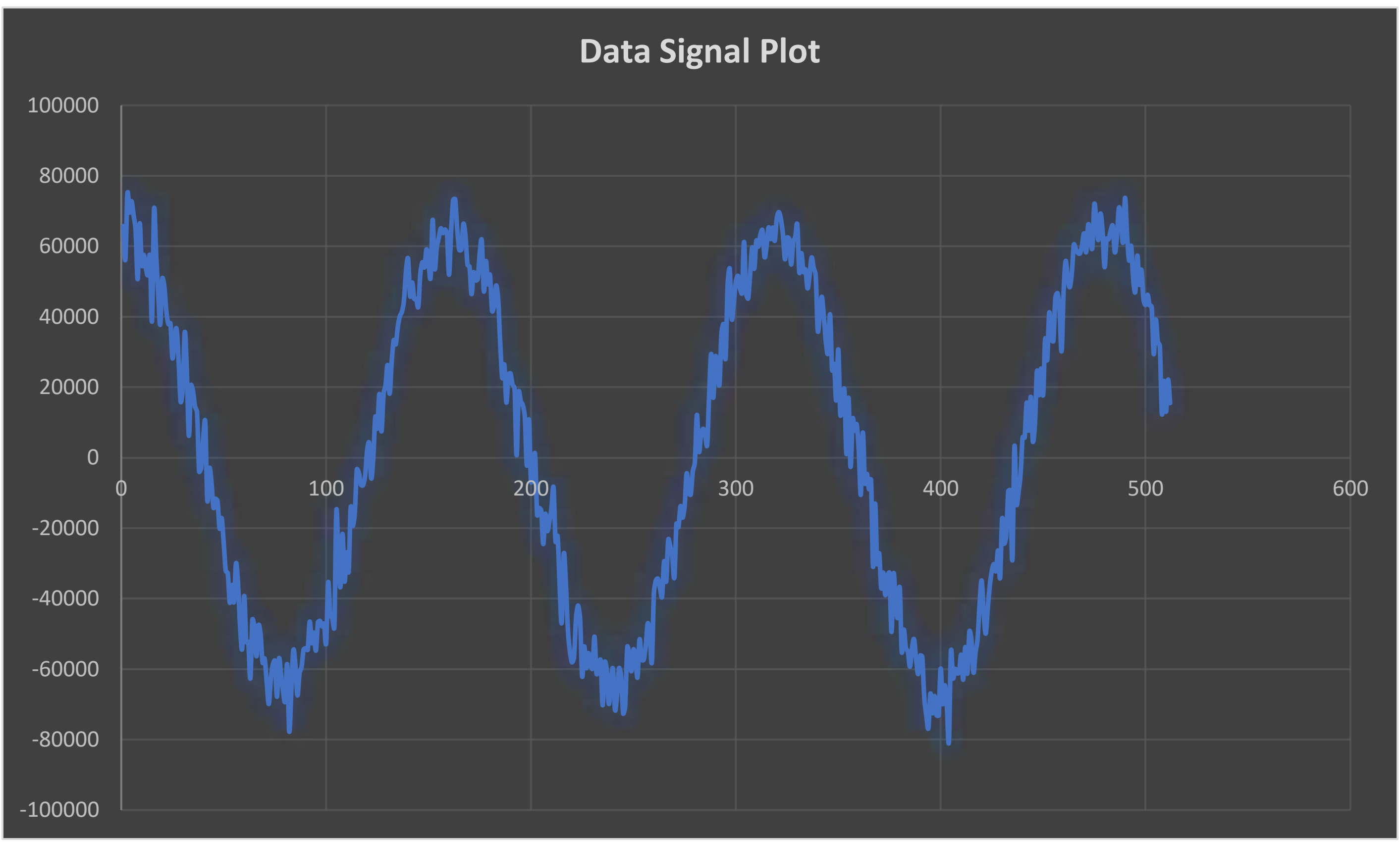
CS-475

Project 7b – written commentary

1. What machine did you run the test on?

I got the best results for OMP and SIMD on the Rabbit server and the best result for OpenCL on the DGX server, so the chart will contain all four tests run on Rabbit and an OpenCL result from DGX.

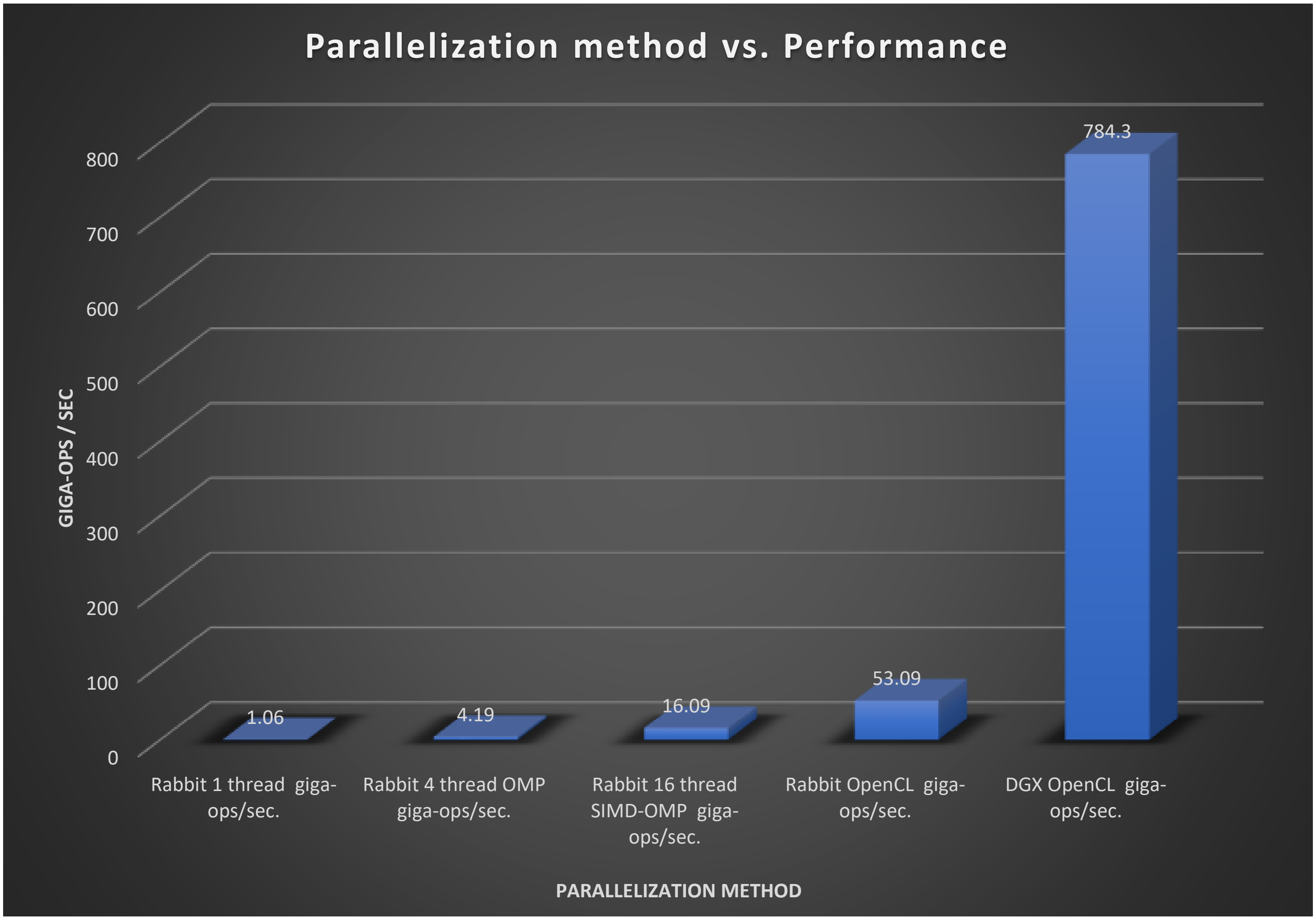
2. Show the sums[1] … sums[512] scatterplot.



3. State what the hidden sine-wave period is, i.e. at what multiples of shift are you seeing maxima in the graph?

The hidden sine wave period is roughly 160 samples between peaks. The graph shows peaks around sample number 3, 163, 321, and 487.

4. What patterns are you seeing in the performance bar chart? Which of the four tests is the fastest, next fastest, etc.? By a little, or by a lot?



The OpenCL performance on the DGX server is far and away the highest performer. It is nearly 800 times faster than the control test with a single thread. The OpenCL performance on the Rabbit server is still more than 50 times faster than the control and many times faster than the traditional CPU based parallelization methods. The second fastest parallelization method is SSE SIMD combined with OpenMP. The four thread OpenMP and SSE SIMD performance shows a perfect speed-up of around sixteen times the single threaded implementation. The four thread OpenMP performance, when compared against the single threaded version, shows a perfect speed up of around four.

5. Why do you think the performances work this way?

The four core OMP implementation is four times faster than the single threaded version because it is able to split the computations onto four separate cores. The version with SSE SIMD added is able to also process four values simultaneously on each core, which leads to 16 times the performance of the single threaded implementation. The DGX server's performance far outpaces the other implementations because the GPU array offers many more cores which can operate in parallel on the problem. The DGX performance is much greater than the performance on the Rabbit server because the DGX server has a larger number of cards allocated to each batch job and the cards being used are more powerful.

I am not sure of why the performance of Rabbit was significantly higher for the three non-Open-CL tests. It seems likely that the CPUs are somehow limited, due to the fact that they are only intended to set up and return data from the GPUs in the DGX array. The DGX array does have OpenMP support, because that is what we are using for timing, so that cannot be cause of it's slowness relative to Rabbit. I am interested in learning more about how I could detect limited performance without having different systems to conduct the tests on, because I fear that a real world system that I could be working on could be hit with a limitation of this type in the future.