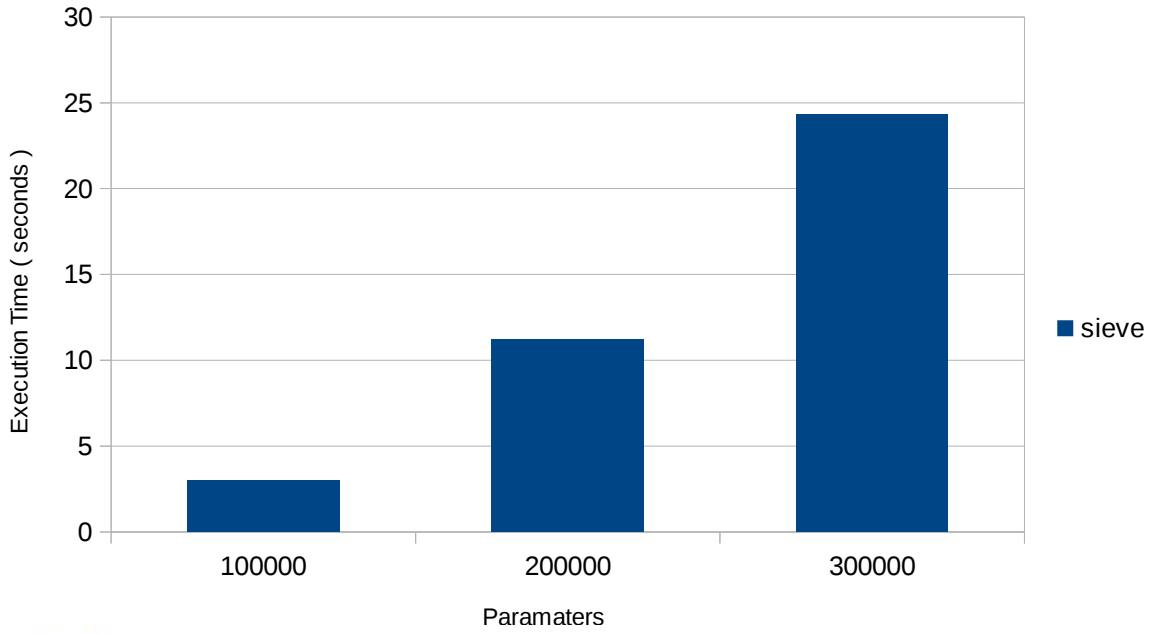
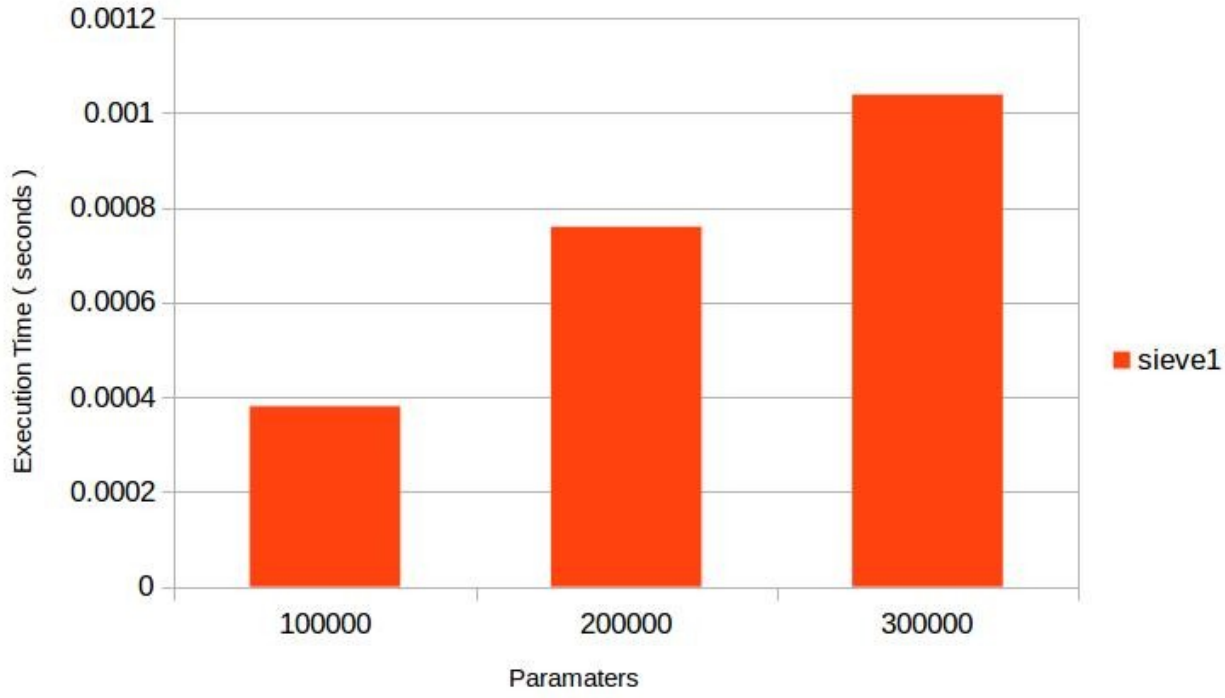
Tanner Pflager
cs475
PA2 Report
9/27/2016

Sieve 1

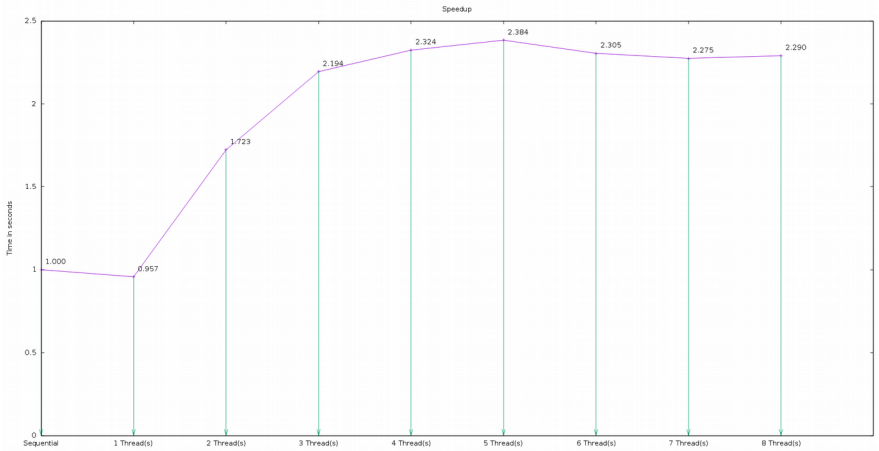
Sieve1 showed much improvement over sieve. Sieve has a complexity of $O(n^2)$. Sieve1 however, has a complexity of O(n/2). As we can see from the graphs below, sieve's execution time grows exponentially where sieve1's execution time grows linearly.



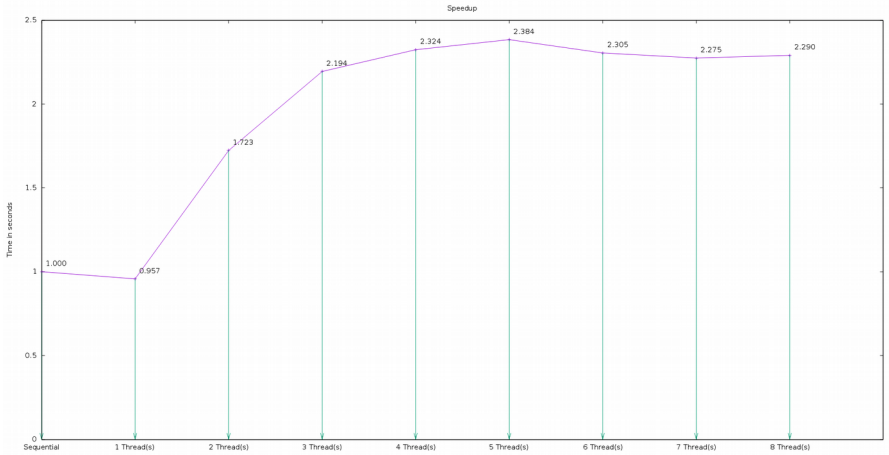


Sieve2

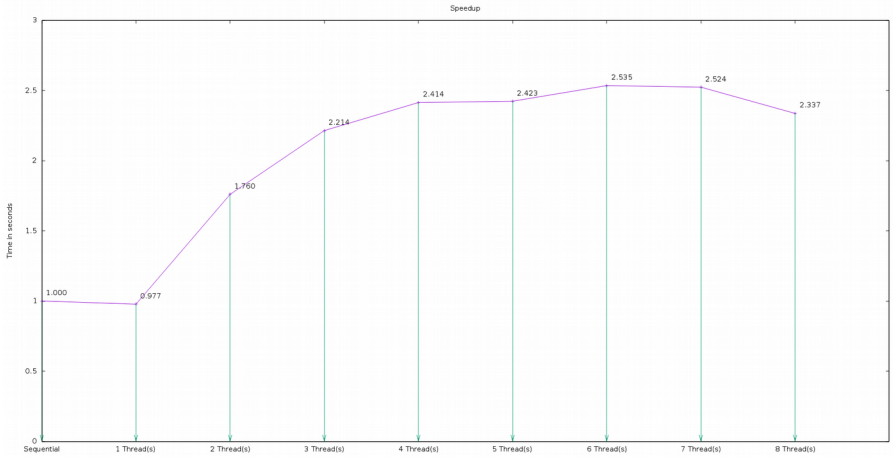
Sieve2 is the parallelization of sieve1. However, due to bad locality, the speedup was not as good as it should have been. As you can see in the graphs (of speedup) for each n (0.5, 1, 1.5 billion), speedup was not linear.



N = 0.5 billion



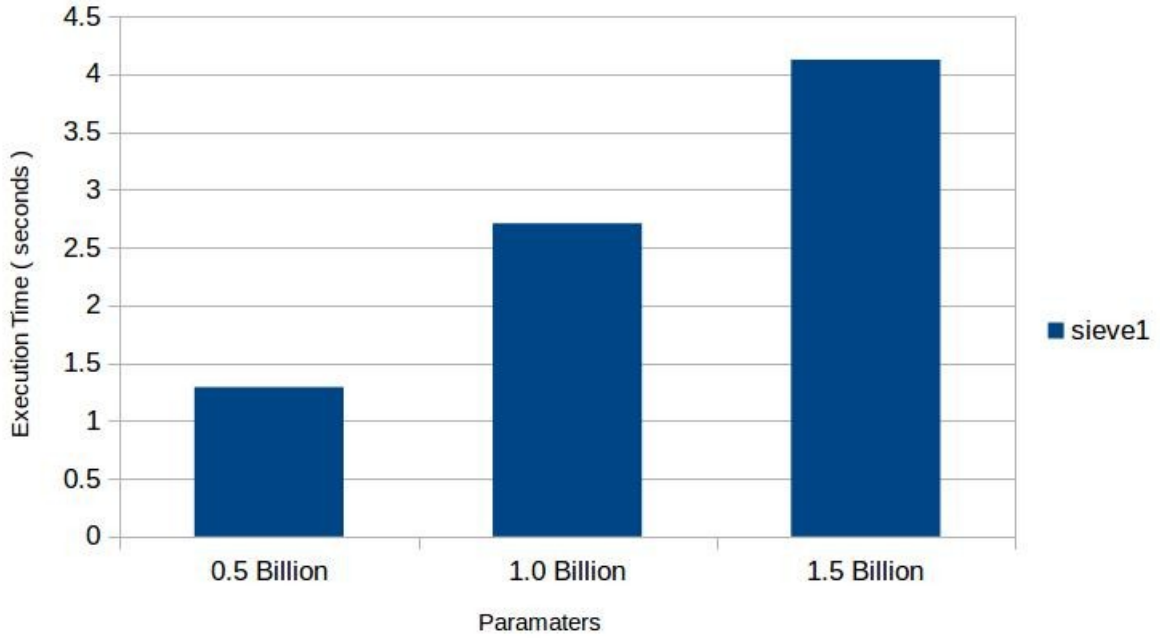
N = 1.0 billion

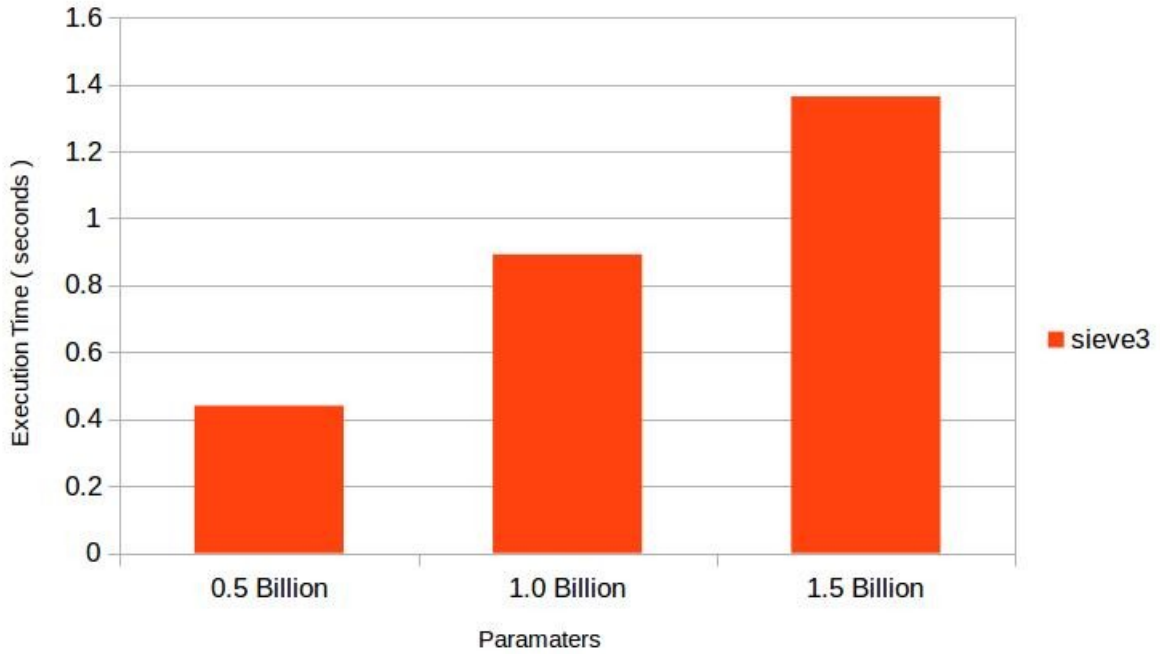


N = 1.5 billion

Sieve3

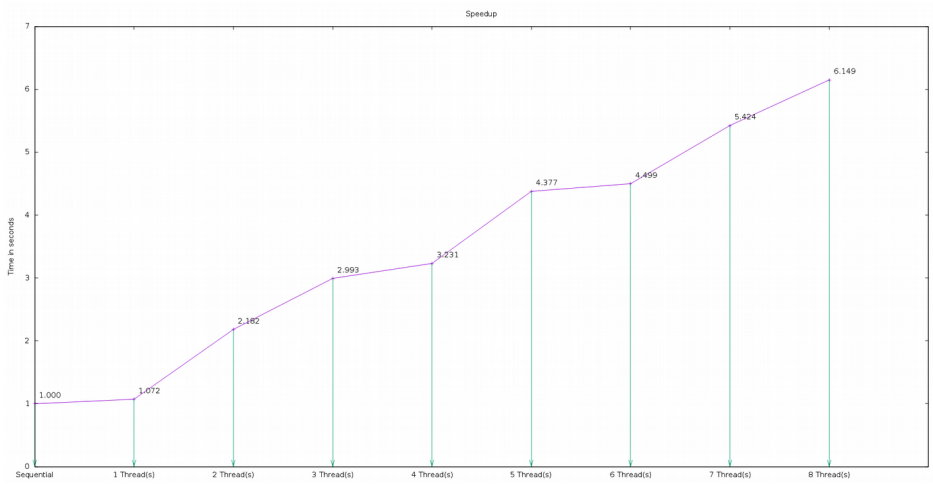
Sieve3 was the solution to the locality problem discovered in Sieve2. The graphs below show the execution time for N = (0.5, 1.0, 1.5 billion). With the block size equal to 1,000,000 I experienced bad performance because the block size was to large to help with cache locality. The block sizes that I found to produce the best execution times for Sieve3 are:
0.5: 300000
1.0: 400000
1.5: 350000



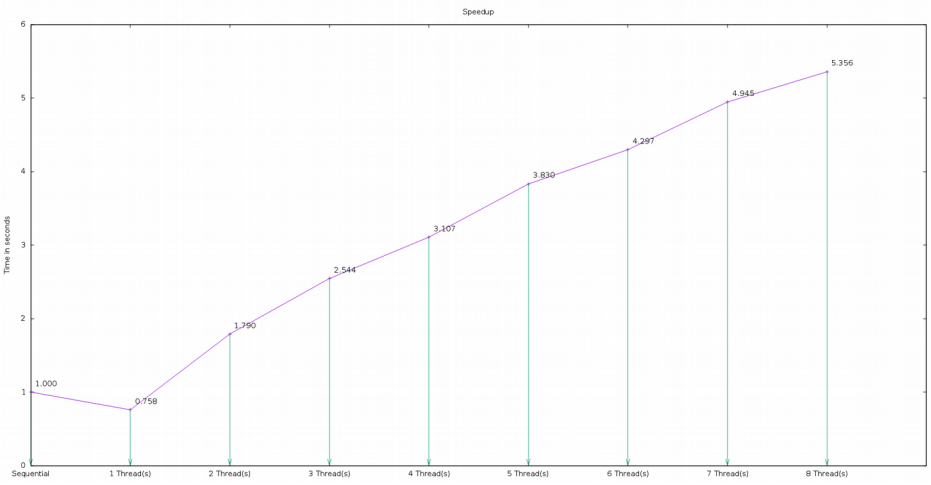


Sieve4

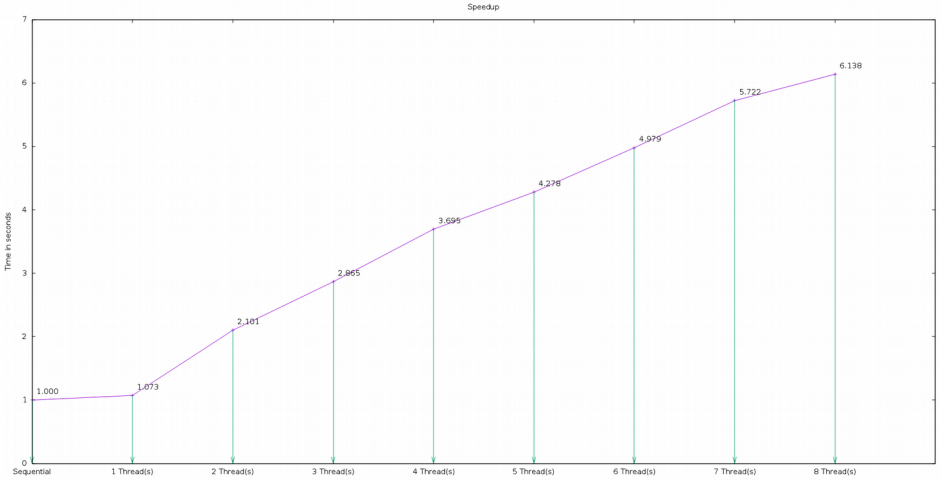
Sieve4 was the parallelization of Sieve3. With Sieve3 fixing the cache locality problems in Sieve1, Sieve4 showed much better speedup the Sieve2. Sieve4 did achieve linear speedup as the graphs below show.



N = 0.5 Billion
blocksize = 300,000



N = 1.0 Billion
blocksize = 400,000



N = 1.5 Billion
blocksize = 350,000