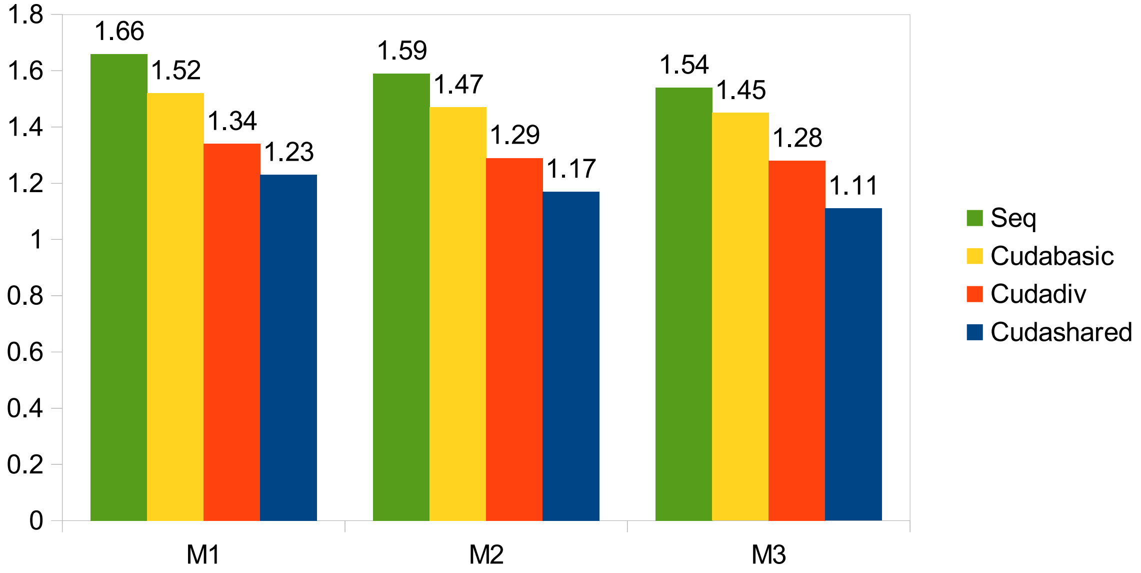
Andrew Gaffney Lab 3 Analysis Parallel Computing



M1 = 60000000 M2 = 40000000 M3 = 25000000

1. The GPU is only better at very large values of M because otherwise the overhead of thread creation and global memory access is too large to overcome the sequential version. At very large values of M, the parallelization gained through the use of the GPU overtakes the sequential version.
2. M2 is lower than M1. The slowdown caused by branch divergence was significant enough to substantially decrease the value of M between M1 and M2. However, the overhead of global memory access and thread creation still leads to a very large value of M2.
3. Similarly, M3 is lower than M2. The repeated accesses to global memory proved a significant hindrance to performance, as can be seen by the substantially lower value of M3. However, as with both previous versions the overhead memory access and thread creation still leads to the GPU only being more effective than the CPU at a very large value of M.
4. I did not implement any other optimization techniques, largely because I could not figure out how to correctly do so. I imagine some would have indeed improved performance, though my inability to implement leads me to be woefully incapable of testing this hypothesis.