
Considering that the fall semester was mostly devoted to researching previous implementations of CUDA and learning to program it ourselves, the spring will be geared more toward writing programs and analyzing hardware optimizations. We are now more comfortable with CUDA, and will move to more complicated programs, and begin to split the workload between teammates.

We intend to program quite a few different programs, some of which include: The Mandelbrot set calculation, different Hidden Markov systems, running Matrix Multiply for larger values, and implementing a Triangular Matrix Multiply program.

Some Hidden Markov models specifically geared toward bioinformatics, such as biological sequencing, gene prediction and protein folding can be implemented by a software package we are looking at called HMMER. We plan to look into parallelizing one or more of these applications using CUDA, and comparing out results against each others’ implementations, the original sequential code, and research that other academic groups have performed using other parallel architectures.

During our research this past semester, we have seen that analytical techniques can be applied to both sequential code, before any parallelization is done, and also to parallelized code to achieve even further optimization. We plan to use the knowledge of these techniques, along with analysis based on hardware specifications (i.e. cache sizes, bandwidths, etc) for additional granularity in our parallelizing efforts.

Another key building block we intend to use to help our project is a very solid timeline. We aim to set more major deadlines, such as a specific date for a program to be completely finished, and also smaller deadlines for program pieces. For larger programs that can be split into sub-parts, we plan to distribute the workload between group members for higher programming efficiency and to complete programs as a faster pace.