Investigating Parallelism in Molecular Dynamics

Abstract:

As an undergraduate researcher for the Dickson lab, I regularly work with software libraries such as OpenMM and Charmm-Gui to submit molecular simulations. Currently, the scale of molecular dynamics is fairly limited; The coordinates of thousands of atoms must be calculated in processes that take dozens of cores weeks or even months to complete. Competing these coordinates involves multiple processors coordinating and communicating with each other. In this project, I will be exploring ways to speed up processes within molecular dynamics, such as the creation and equilibration of environments, forming submission scripts, or explore the far more complicated topic of the simulations themselves. Hardware used will be the HPCC and the in-house system in the Biochem building, called “Markov”

Part 1:

As my understanding of OpenMM is somewhat limited and in need of exploration, most of what I plan on doing for the first few weeks is exploring in detail the mechanics behind molecular dynamics simulations and the process behind submitting scripts. It is possible that the timesaves I am capable of building will revolve less around the simulations themselves and more so the process behind building/submitting submission files and processing the results.

By the end of this step, I expect to have identified a bottleneck in the process of submitting, running, or processing code and files in molecular dynamics and have devised a system for evaluating the code with specific sets of circumstances (e.g. environment, specific protein to model, ion/waterbox constraints etc.)

- The expected process I have for performing benchmark tests is to come up with 3-4 simulations with predetermined run lengths, procedures, and constraints
- As there are multiple processes related to setting up and submitting up runs that (to my knowledge) cannot be measured with time commands, I will have to set up a process to manually record the times it takes to perform each run

Part 2:

Once I have identified a piece of code, I will be exploring different methods to parallelize or otherwise speed it up. Success will be measured by 1. Recording the avg. time it takes to run jobs in the way I have specified in part 1. And 2. Seeing if the code or process I have jotted down is capable of handling different kinds of processes (e.g. submitting variable types of molecules for analysis, asking for/running extensions of existing code, variable numbers of substeps .etc)

Schedule:

- Feb 10-16: Begin research, investigate OpenMM documentation and Molecular Dynamics
- Feb 17-23: Continue research, explore coding elements of MD simulations
- Feb 24-Mar 2: Turn in part 1, by this point a bottleneck and segment of code to improve should be identified
- Mar 3-9: Explore and write down possible avenues of improvement
- Mar 10-16: From here on out it is just working with the code, seeing what I can do, constant tinkering and headaches based on circumstances I cannot predict
    - Mar 17-23
    - Mar 24-30
    - Mar 31-Apr 6
    - Apr 7-Apr 13
- Apr 13-15: Finalize project, turn in