Optimizing a Python-based Neutron Spin Dynamics Simulation Code

Chris Swindell, TTU Computer Science

The UCNτ experiment finds the mean lifetime of an ultracold neutron (UCN) trapped within a magnetic field before it undergoes beta-decay. One piece of information that helps determine this mean lifetime to a high precision is a neutron's depolarization lifetime (the average time from when any given UCN enters the field polarized until exiting via depolarization). One simulation we are currently developing attempts to determine this value by calibrating against trap lifetimes measured at different holding fields, which requires large amounts of mathematical computation. On a 10-core computer running Linux OS (at 1 neutron per core simultaneously), this currently takes about 36 hours to complete when simulating 100 neutrons. Realistically, we would want to run around 1,000,000 neutrons in a simulation to reduce our margin of error; however, that would take over 40 years to complete on the same hardware when scaled up. To reduce the amount of time it takes to run a simulation, optimization of the simulation code is required. To fulfill this requirement, this simulation code (originally written in Python) was benchmarked and optimized before conversion of part, and potentially all, of the code to C++. The methodology followed and what was found to be good practices for code optimization in general throughout this process will be presented.