FAST N-BODY SIMULATION AND DISPLAY

Kade Phillips and Scott McCuen

Our project is a demonstration of an FPGA's strengths as dedicated, parallelized, application-specific hardware. An interesting computation that showcases these properties is direct N-body simulation. On a general purpose processor, this has a time complexity of $O(n^2)$, but on an FPGA this can be done with a time complexity of $O(n)$ for reasonable n (less than 4000 on a Virtex-7). We expect that our FPGA implementation will outperform an equivalent program in C running on a mid-range processor by one to two orders of magnitude for $n > 1000$. FPGAs seem to be used frequently as accelerators for N-body simulation, but we are not aware of an FPGA being used for the computation in its entirety.

The project is divided into two sequential modules: calculation and display. The simulation takes place in a box one million light-years across, with 1000–4000 particles. There is one register+arithmetic unit per particle. Wiring each unit to every other unit would reduce the problem to constant time but this requires impossibly dense routing. Our solution is to use a serial bus that communicates the position of each particle every timestep. To reduce the footprint of each unit (and headache for us), we expect to use fixed-point operations instead of floating-point. With a 100 MHz clock, we expect to run at ~100 timesteps per second.

We will display the position of particles at 30 frames per second, combining adjacent particles into groups and setting pixel brightness based on the density of particles in a group. At a minimum, we will be able to view the cube from the top, front, and side. If sufficient resources remain on the FPGA, we will add features to rotate and zoom in on the cube.