

Project Checkpoint

Updated Schedule

[Schedule | GPUFluidSimulator \(oadrian.github.io\)](#)

What we've done so far:

So far, we have a working basic implementation of a sph fluid simulation. To start, we implemented the N^2 algorithm taking into account the densities and forces from all other particles in the simulation. Doing this sequentially as well, the size of the simulation we could run was very small (as expected). Next, we added both parallelism through openmp pragmas and additional acceleration with z-indexing. For z-indexing, the simulation box is divided into sections, then the particles are placed in a box, and sorted according to their boxes bit-interleaved 3d coordinates. With the particles sorted and placed in boxes, each box can run in parallel and only worry about density and force calculations for the particles it contains. Due to sorting by box order, there is good spatial locality for the memory access to the particle array.

How we are doing:

We are on track with our original checkpoint goal of having a working CPU simulation, but behind by a week getting the GPU solution started and working. Upon further consideration, the GPU solution naturally arises from the CPU one and should not be too difficult to implement now that we have the CPU based one working. (The GPU just allows a similar approach to be taken while offering parallelism over the particles in the box as well). The fluid surface rendering still is a stretch if we run into significant hiccups while implementing the CUDA kernels. One of the reasons we got set back was that we spent some time in the beginning trying to write our own OpenGL solution for visualizing the particles, given that neither of us had real experience using OpenGL and the level of quality that we wanted from the graphical representation we opted to use the graphical solution for NVIDIA's Particle Demo. Another challenge when writing this simulation has been the parameter tweaking for the physics formulas.

Poster Session:

We plan on showing both a live demo of our simulation as well as graphs comparing performance.

Preliminary Results:

We have a GIF on our website showing our simulation in action.

[Project Checkpoint | GPUFluidSimulator \(oadrian.github.io\)](https://oadrian.github.io/Project%20Checkpoint/GPUFluidSimulator)

Updated Goals:

- Making sure that number of particles in a grid slot is capped
- Debug openmp particle collisions
- Write cuda kernels
 - Density
 - Pressure
 - Force
 - velocity
 - Position integration
 - Z-index sorting
- Implement cuda kernels
- Benchmarks
- Create demo scenes (if time allows)
- Fluid surface rendering (if things go smoothly)

Concerns:

- Parameter tweaking: it seems like our simulation is going to entirely depend on what parameters we choose and given that neither of us have a solid background in hydrodynamics this is a bit more challenging than anticipated. There are several resources online but they each approach the problem differently and thus when it comes to the parameters they vary widely. We've tried to stick to a couple of resources/tutorials but sometimes they don't have enough information for how we want to solve the problem. If we want to get the fluid surface rendering done as well it would be a lot of OpenGL work beyond rendering simple dots/spheres in space.