

Timothy Parsons

Brown University Computer Science Sc.B. '14 Capstone Abstract

GPU-Accelerated Material Point Method Snow Simulation

For my capstone project in CS224 (Interactive Computer Graphics with Professor John Hughes) a group of four students (Eric Jang, Max Liberman, Wil Yegelwel, and myself) decided to implement Disney's Material Point Method for snow simulation. Their paper can be seen [here](#), a video of their results can be seen [here](#), and of course feel free to reference their latest animated film Frozen for more examples. The main idea of the simulation is that this is a grid-based particle system where at each time step the snow particles are rasterized to a Cartesian-based grid, calculations are done on the grid to account for collisions and forces, then the particles' positions and velocities are updated based on these calculations. To extend on Disney's work, which was done all on the CPU and resulted in long simulation times, we decided to implement the simulation on the GPU to try and achieve some aspect of real-time simulation as well as speeding up simulation times in general.

The overall project involved programming the simulation steps in CUDA, coding up the user interface, and writing an offline renderer to get some pretty results. I specifically worked on a few parts of the project: I helped with initial/ongoing design, wrote up CUDA code for rasterizing particles to the grid, and worked on aspects of the user interface. Regarding the user interface I worked mostly on aspects that had to do with implicit colliders. We supported sphere and half-plane colliders that the snow particles could interact with. I implemented the ability to render them, unified logic between our simulation logic and on-screen rendering logic, and I helped a little with being able to save them and load them in a scene. I also implemented a velocity tool that could give a collider velocity as well as a fillable mesh (which could be any .obj mesh that could then be filled with snow particles), which then would be transferred to the snow particles after filling. The direction and the magnitude of the velocity could be controlled by rotating an arrow or dragging it to change its length.

Below you can see a picture of our user interface and a snapshot of a snow castle being collided with a sphere collider. You can also check out some of our scenes in a video [here](#) that Eric Jang put together. Enjoy!

