Joseph Simons
CS497 Abstract

The problem that we are tackling deals with rendering gaseous phenomena in real-time, in such a manner that it will be easy for an artist to control and shape. Up until now, all gaseous modeling techniques deal with either procedural or physics-based simulation algorithms. However, this is very difficult for artists to work with since they have to understand and manipulate several complicated variables in order to achieve what they want to see. Even the tiniest error can quickly grow into an artifact that distracts from their intended results. There has been research into developing an approach using keyframes to model smoke simulation; however, this is an offline process that may take several hours for a simple animation. Our priorities include visual realism, interactivity through an intuitive mouse-driven interface, and scalability to production offline renderers.

In modeling natural phenomena, artists compromise the benefits of controlling features directly with the realism of simulation. Our hybrid approach attempts to provide familiar modeling tools for a controllable gas simulation. The virtual gas exists both as a medium resolution simulation and procedural primitive abstraction. The user can manipulate either representation and simultaneously modify the other, useful to directly guide a simulation's advection, or more naturally evolve a procedural model. We adapt simulation and rendering techniques across the CPU and accelerated graphics hardware for an interactive environment. Specifically, coarse feature recognition quickly evaluates a simulation basis, an enhanced implementation of advected textures conveys moving media, and a processor-distributed renderer balances processing between the CPU and graphics hardware. The artist can use these, as well as our already implemented density primitives, to define the keyframes that the flow will follow. Noise is used to generate high-resolution detail, and we use advected textures to move this noise to simulate and render flow. We have modified the existing technique to continuously adapt strain, versus the original blending approach used due to the errors and inconsistencies that may arise. The resulting system produces realistic gas animations under artist control in an interactive environment.

So far for results we have implemented realistic flow in two dimensions, and are working on extending it to three. We are using the existing framework from our previous project, so we already have procedurally defined clouds that are manipulative. We also have basic procedural flow working as well, though we are working on adding tool support through Houdini, a commercial rendering and modeling program, in order to allow the artist to control and define features without having to manipulate the code base. We have basic feature recognition working with simple vortices. This still needs to be finished and fleshed out. Everything is proceeding smoothly thus far, but we are working from opposite ends are going to attempt to make it work when we reach the middle. This may cause problems and setbacks if we do not plan or communicate properly. We are also running into problems concerning CPU power, and may have to skimp a little on the 3D flow, or modify it to appear 3D while only being 2D, in order to compensate for existing computational power in desktop computers.

I am carrying out this project under the supervision of Prof. David Ebert (ECE, Purdue) and in collaboration with Joshua Schpok (ECE Graduate Student, Purdue).

Previously, I have worked with Josh to get the initial procedural flow working, and getting our new advected texture approach to work. I have also helped design the approach that we are currently using. I have gotten basic tools with Houdini to work and define the position and size of the cloud/flow bounds, and am currently working on getting flow working by using the tools already implemented in Houdini. This complements the work that Josh has done on 2D flow, and basic feature recognition.