Particle Systems Using CUDA

Mary Haugh
Background

- Particle systems are essentially just a lot of particles moving in a certain way.
- There's a lot of uses for them in gaming (fog, smoke, snow, etc.)...although this program doesn't really have such applications.
Demo

• What You Will See
  – 50K (or more, if one doesn't mind a slightly slower animation) particles moving in patterns on a screen
  – Beneath the particles are their shadows

• This Will Need Ethernet...
Particle Movements

• The Physics
  – The particles are attracted to a point on the screen.
  – A particle's distance from this point determines the “force” acting upon it.

• The Math
  – Every call to display() updates the point according to a Bezier curve
  – Every so often, in order for the system not to stabilize (making the animation boring), a “shock” is randomly added where the point changes its position by a large amount on the screen.
  – For the swirly effects, the “point” is actually a “circle” that will push the particle away with a larger force if it moves inside of it.
Shadows

- Originally, this was a 2D system for CS179.

- Conversion to 3D
  - The attempt to turn the particles into spheres resulted in a really slow system (unless one uses about 3k particles, which isn't very impressive)
  - Added a third dimension to the program – requires carrying around another array of position coordinates
  - Translate these positions onto the “floor” in the program
  - The particles are technically still rendered 2-dimensionally (we're still using GL_POINTS, whose movements the third dimension affects to a very small degree)
  - However, the shadows are based on this third dimension which is what makes it realistic.
The program uses OpenGL, which renders spheres very slowly. So we use GL_POINTS. With only the CPU, there would be so much lag in the program at about 5k. Integrating with CUDA allows us to display up to 100k particles. Probably more actually. Since we're running this off of servers and displaying things on our screen, we're most likely limited by bandwidth rather than by program speed.
Future Directions

- The shadows look a little funky right now because 1) their transparencies don't change with height from the ground and 2) their edges are hard.
- It would be nice to convert this application into a screensaver...