Parallel Fluid Simulation
15418 Final Project

Haoying Zhang (haoyingz), Qilin Sun (qilins)

November 13, 2023

1 Summary

Fluid dynamics is famous for its mathematical complexity, which is also why computer scientists fascinated about it. In this project, we will model an amount of fluid particles and allow the user to add, disturb, or remove the particles, and reshape the container. We will use CUDA to speed up the simulation.

2 Background

To reassemble realistic fluid, a large amount of particles are created, and they are smoothed to approximate the continuity of real fluids. The technique to do so is known as Smoothed Particle Hydrodynamics, where each point mass is viewed as a continuous distribution, and the property of the fluid at any location will be a weighted sum of nearby particles' properties. The further the particle is from the location, the lower its contribution is.

The motion of individual particles is driven by the pressure at its location, which points in the direction of decreasing density, and whose magnitude is proportional to the different between the current density and the desired density (defined to be the average density of all particles over the space). This is based on the intuition that the system tends to transition to a state of uniform pressure.

3 The challenge

Because computations are always performed on a point-by-point basis, the simulation process is highly parallelizable. In addition, a source of inefficiency is the same one encountered by Assignment 3/4, which is not every particle needs to be considered for every other particle. Particles that are outside of the domain of influence do not need to be considered. That is, methods similar to blocking would be necessary to significantly enhance the performance.

4 Platform Choice

We will first implement a CPU version, then a CUDA version. Because the simulation is computation-intensive, GPU may be a good choice, but only if the implementation is wise enough.

5 Resources

This article from Nvidia discusses a way to split particles into groups. This paper introduces a way to maintain particle consistency.

6 Goals and Deliverables

Goals:

-
- Compare the performance of CPU and GPU implementations.

Hope to achieve:

- If it goes well, achieve at least 4x speedup from CPU to GPU implementations.
- If it doesn't, 2x speedup is enough.

Demo:

- Play a recorded animation of the simulation of 10k particles in a finite space.
- A

7 Schedule

Week	Plan
Nov. 13 - 19	Background research and initial code-up
Nov. 20 - 26	Refine CPU version, finish CUDA version
Nov. 27 - Dec. 3	Refine CUDA, benchmarking
Nov. 4 - Dec. 10	Documentation and presentation preparation

Table 1: Schedule