Creating and Optimising a Fluid Simulation For Use in Video Games

Sakir Azimkar
200823588

Supervised by
Richard Davison

Problems

- Computational Fluid Dynamics are integral in many industries due to the sheer presence of fluids in everyday life.
- Existing fluid simulations require a lot of processing power and are expensive to obtain.
- This results in a lot of video games to useshaders that look like fluides instead of implementing actual fluid flow.
- Most commercial CFDs do not allow viscosity configuration as they are tailored specifically towards water.

Solution

A free, open-source project that utislises the Navier-Stokes equations to produce simulations for fluids of any viscosity with an emphasis on performance.

Aim

To develop a fluid flow simulation for fluids of all viscosities and attempt to optimise it to improve performance in video games.

Objectives

- Explore existing fluid simulations and identify issues with them from the perspective of a game developer.
- Understand and explain the importance of the Navier-Stokes equations on viscosity.
- Learn and utilise a new API that can be optimised with more freedom than the pre-existing game engine.
- Develop a basic 2D fluid simulation that will then be extended to 3D.
- Test and evaluate which elements of the simulation affect performanee the most and attempt to optimise these specific parts the most.

Technology

I will be using OpenGL, a cross-platform, cross-language graphics API to render the fluid. Since OpenGL is C-based, I chose to use C++ as the programming language. C++ provides the additional benefit of object-oriented programming, which allows for better structure and scalability in the project.

Features

- Adjustable viscosity.
- Real-time rendering and visualisation.
- Smoothened Particle Hydrodynamics (SPH).
- Utilisation of the Navier-Stokes equations, listed below as figure 1 and figure 2.

Future work

- A table or UI that allows the user to adjust values in real-time.
- Preset fluids, such as water, syrup and slime.
- Implementing the simulation into a real game and testing how the performance is affected.
- Further optimisation

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} + v\nabla^2\mathbf{u} + \mathbf{f}$$

Fig 1: Navier-Stokes equation for incompressible flow of a Newtonian fluid with constant viscosity

Change in velocity over time = advection + diffusion + body force

$$\frac{\partial \rho}{\partial t} = -(\mathbf{u} \cdot \nabla)\rho + k\nabla^2\rho + S$$

Fig. 2: Navier-Stokes equation for density moving through the velocity field

Change in density over time = advection + diffusion + external force

Advection = transportation of fluid properties due to its flow or the force exerted on a particle due to its surrounding particles.

Diffusion = "damping" caused by a viscosity or density constant.

Body force = external forces that act on the fluid density such as gravity, wind or smoke from a fire.

Stam, J. (n.d.). Real-Time Fluid Dynamics for Games. [online] Available at: https://damassets.autodesk.net/content/dam/autodesk/www/autodesk-reasearch/Publications/pdf/realtime-fluid-dynamics-for.pdf [Accessed 24 Mar. 2025].

Mia John (2011). Animation of Fluids - SlideServe. [online] SlideServe. Available at: https://www.slideserve.com/Mia_John/animation-of-fluids [Accessed 25 Mar. 2025].