1 Research Review

Much of my research comes from GitHub, a Microsoft owned cloud-based platform for developers to store their personal or professional projects and publish them for wider use by the community. From the GitHub page of Sebastian Lague, I was introduced to articles on SPH written by many reputable institutes such as by *ETH Zurich*, *Université de Montréal* and *University College London, UK*. The majority of these papers had affiliation with *SIGGRAPH*, the international Association for Computing Machinery's Special Interest Group on Computer Graphics and Interactive Techniques. SPH related techniques are researched and published most for showcase at *SIGGRAPH* events. Through the use of google scholar and *SIGGRAPH*, I have been able to narrow my search for related documents for simulating liquids.

1.1 History and Relevant Literature

Lucy [**?**] introduced Smoothed Particle Hydrodynamics as a numerical testing tool for astrophysical calculations involving fission[1] within stars. This idea of quantity interpolation or "approximation" of fluid quantities was furthered by Gingold and Monaghan [**?**] and applied to non-spherical stars. Although both sources provide appropriate applications of this technique, the obvious limitation is that the majority is within the context of Astrophysics and not CFD. Additionally, both sources were released in 1977 with major development in the simulation field, such as the use of more modern optimisation techniques which utilise the powerful hardware now widely available, leading to the source being obsolete for present-day applicational use.

The work of Müller *et al.* [**?**] adapted SPH for interactive fluid applications, the first of its kind, putting forward an alternative Lagrangian method than the more common Eulerian method used for CG and modelling purposes. The paper provides a gentle introduction to SPH with a mathematical brief to the most important phenomena observed within fluids for simulation, including Pressure, Viscosity and Surface Tension. There is a distinct lack of algorithms, which leaves implementation up to the reader but the paper fulfills its purpose as an excellent introduction to SPH.

After the foundational work in 2003, Clavet *et al.* [**?**] release their work two years after with the primary focus on implementation, introducing key algorithms such as the Simulation step which covers the pseudocode for every frame of the animation and how the quantities of individual particles change frame by frame. Problems specific to implementation are also acknowledged, for example the near-density and near-pressure tricks are also introduced which prevent an issue that causes liquid particles to cluster.

An example of a more recent publication is Koschier *et al.* [**?**] in 2019. This tutorial summarises the state of SPH in its entirety by covering the theory and implementation rigorously, but also with a focus on optimisation methods to lessen compute time utilising modern hardware. The tutorial is diagrammatic and visual helping reinforce the ideas being expressed. Compared to earlier iterations covering SPH, this paper acts as the

[1]Splitting of atomic nuclei causing a release of energy

ultimate guide by placing all the information needed in one document. The paper dives much deeper into the niche complexities involved with simulating any fluids or even soft-bodied solids, but are beyond the scope of this project.

1.2 Alternative Approaches

An alternative approach for simulating fluids I have mentioned across this write-up is the Eulerian approach. Robert Bridson, in his book *Fluid Simulation for Computer Graphics* [**?**], perfectly encapsulates the Eulerian Viewpoint. In his words, *"The Eulerian approach, named after the Swiss mathematician Euler, takes a different tactic that's usually used for fluids. Instead of tracking each particle, we instead look at fixed points in space and see how measurements of fluid quantities, such as density, velocity, temperature, etc., at those points change in time. "*. The non-particle centric approach means the fluid is treated like a continuous medium and the simulation solves Partial Differential Equations (PDEs) to model its behaviour. Spatial domain is split up into equal sized grids and the fluid is modelled as being incompressible which would mean the total inflow within a grid must equal the total outflow. This process leads to advection within the simulation and then is rendered on screen.

Grid-based approaches have the advantage of having higher numerical accuracy and efficiency because solving PDEs can be optimised using techniques like finite difference or finite volume methods. Exactly enforcing incompressibility is important for accurate production of turbulence, and SPH methods have a hard time enforcing incompressibility efficiently. They also can have difficulty allocating computational elements throughout space efficiently. For these reasons, they have not been demonstrated to be effective for calculating flows such as air around a car, as stated per this paper by NVIDIA employees in proceedings of the 2010 ACM SIGGRAPH symposium [**?**].

For my artefact, I aim to create a semi-realistic animation of a liquid suggesting SPH is a viable technique as I do not aim for accuracy like some applications in industry require, for example modelling airflow around rocket fuselages. Furthermore, I aim to create an element of interaction by resizing windows to show some kind of advection. "Particle-based methods like Smoothed Particle Hydrodynamics (SPH) are attractive because they do not suffer from the limitation to be inside a box" [**?**], also implying that resizing is impractical to implement with an Eulerian approach as resizing the window would restructure the grids that an Eulerian approach relies upon.

1.3 Software