Project Proposal: Investigating Real-time Fluid Dynamics Systems

Context:

Smoothed Particle Hydrodynamics (abbreviated to SPH throughout this document) is a powerful computational technique for simulating fluid dynamics. It can be used to simulate complex fluid phenomena with high fidelity, making it useful in many fields such as engineering, scientific research, and potentially interactive simulations in gaming environments.

However, due to its high fidelity, it also comes with a large computational load, so there is a need to optimize its computational efficiency on both CPU and GPU architectures. Traditionally, SPH simulations often rely on CPU-based implementations that may struggle to deliver the necessary performance for real-time applications such as games, particularly as particle counts increase. By leveraging the parallel processing power of modern GPUs, we can diminish these issues, although this introduces many challenges and considerations.

Therefore, the overall goal of this project is to investigate and address many of the challenges associated with optimizing real-time fluid dynamics simulations using SPH on both CPU and GPU architectures. To achieve this, I will produce a real-time fluid dynamics simulation in Unity 3D, which will implement both CPU and GPU-based SPH approaches and analyse the performance and scalability as the particle count of the simulation increases.

|  |  |
| --- | --- |
| **Survey on SPH Methods in Computer Graphics** [1] | **Description**: This survey offers an overview of prevalent methodologies within the realm of Smoothed Particle Hydrodynamics (SPH). Additionally, the paper provides a comprehensive overview of the fundamentals of the SPH formalism, including discussions on kernel functions.  **Relevance**: Given the focus of my current project on SPH and performance optimization, understanding prevalent methodologies is essential. Furthermore, this paper serves as a gateway to further exploration in the topic area by providing references to more in-depth sources for additional research. |
| **SPH Fluids in Computer Graphics** [2] | **Description**: This report provides a summary of research within the area of Smoothed Particle Hydrodynamics. The paper mentions many different approaches and methods in related areas of SPH, for instance, approaches to estimate the neighbourhood of a particle, and section 1 introduces the underlying fundamentals of SPH.  **Relevance**: This paper is directly relevant to my current project focused on SPH and performance optimization, particularly concerning neighbourhood search algorithms. Despite some overlap with “Survey on SPH Methods in Computer Graphics [1]”, this paper offers additional insights into the area of neighbourhood search. However, its relevance is slightly diminished by its older publication date in comparison to [1]. |
| **Optimized Spatial Hashing for Collision Detection of Deformable Objects** [3] | **Description**: This paper presents an optimized approach to collision detection for deformable objects using spatial hashing. It outlines methods to enhance performance, including the use of prime number hash table sizes.  **Relevance**: As "SPH Fluids in Computer Graphics [2]" highlights the significance of spatial hashing for neighbourhood searches in SPH, this paper provides useful context for this approach. Also, it offers insights into optimizing spatial hashing, aligning with my project's objective of performance enhancement in SPH simulations. |
| **Particle Simulation Using Cuda** [4] | **Description**: This document describes how to implement a simple particle system in CUDA, NVIDIA’s parallel computing framework for NVIDIA GPUs.  **Relevance**: In the context of my project, it offers valuable insights into parallelization techniques on GPUs, which are crucial for optimizing the performance of my Smoothed Particle Hydrodynamics simulation. While my implementation may utilize compute shaders in Unity 3D for atomic operations, understanding CUDA's principles enhances my ability to leverage GPU parallelism effectively in achieving real-time SPH simulations. |
| **Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments** [5] | **Description**: This paper provides an overview of smoothed particle hydrodynamics and, in particular, contains a detailed exploration of smoothing kernel functions within the context of smoothed particle hydrodynamics (SPH).  **Relevance**: In the context of my project, it is important that I am able to choose a kernel function based on computational efficiency and accuracy. This paper provides insights into helping me select a kernel that prioritizes computational efficiency over accuracy while ensuring a minimum level of accuracy is maintained. |
| **Smoothed Particle Hydrodynamics and Magnetohydrodynamics** [6] | **Description:** This paper presents the mathematical basis for smoothed particle hydrodynamics including how many of the equations in smoothed particle hydrodynamics are derived and shows how to interpret the equations.  **Relevance:** This paper is relevant in the context of my project as it assists me in understanding the underlying mathematical basis of SPH. This understanding will help my project to achieve accuracy in the calculations and increase the overall quality of my implementation. |

Aim and Objectives:

**Aim**: To analyse and optimize the performance and computational efficiency of real-time fluid dynamics simulations using Smoothed Particle Hydrodynamics (SPH) on both CPU and GPU architectures.

**Objectives**:

1. Identify key features crucial for accurate fluid simulation using Smoothed Particle Hydrodynamics (SPH), while emphasizing simplicity to optimize performance. (e.g. surface tension).

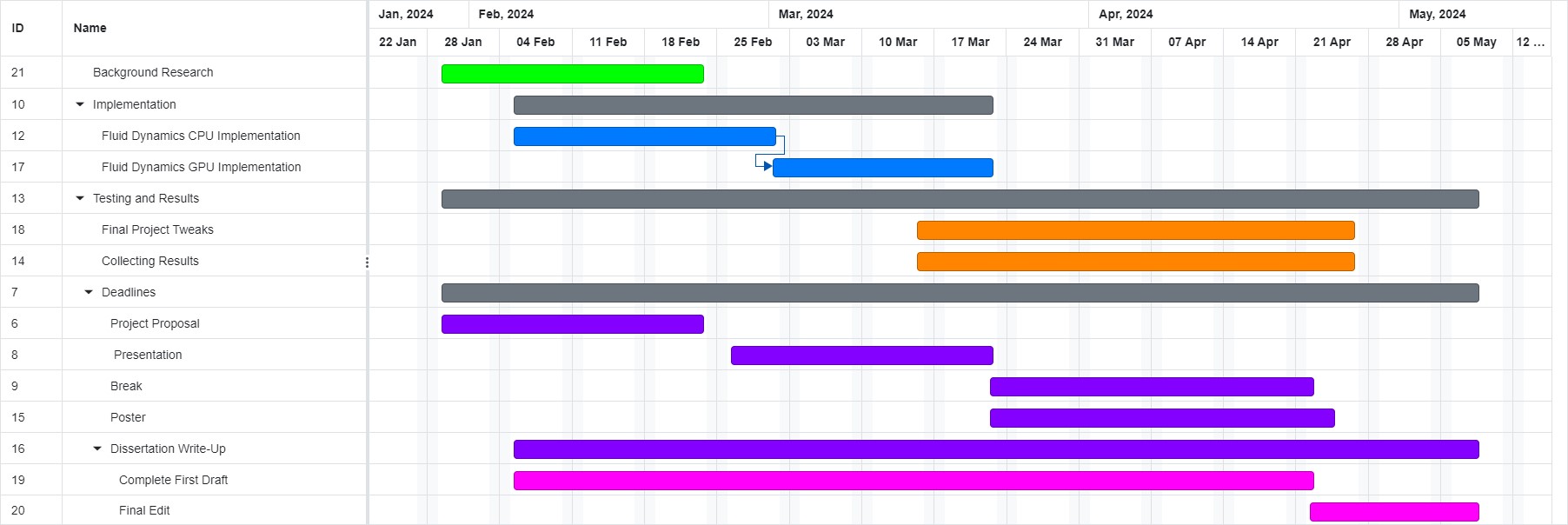
2. Research existing Smoothed Particle Hydrodynamics (SPH) based fluid dynamics simulation techniques to enhance performance and determine the most effective approaches to employ within the timeframe of the project.

3. Implement the researched techniques and identified features to develop a fluid simulation system using Smoothed Particle Hydrodynamics (SPH).

4. Implement parallelized SPH algorithms tailored for GPU architectures to leverage their parallel processing capabilities and optimize memory access patterns.

5. Analyse the scalability of the optimized SPH simulations across varying particle counts, examining how memory usage and computational demands scale with increasing particle count.

6. Compare the performance of CPU and GPU implementations in simulating fluid dynamics across varying particle counts in terms of framerates.

Planning:

*Figure 1: Project Gantt Chart*

I spent the first week, starting January 29th, defining my project theme and conducting preliminary research. From the second week, starting February 5th, I began the implementation process. To ensure a systematic approach, I divided the implementation stage into CPU and GPU segments, allocating more time to the CPU implementation as it serves as the project's baseline. Despite initially allocating research time until mid-February, I will continue research alongside implementation, focusing on specific aspects as needed.

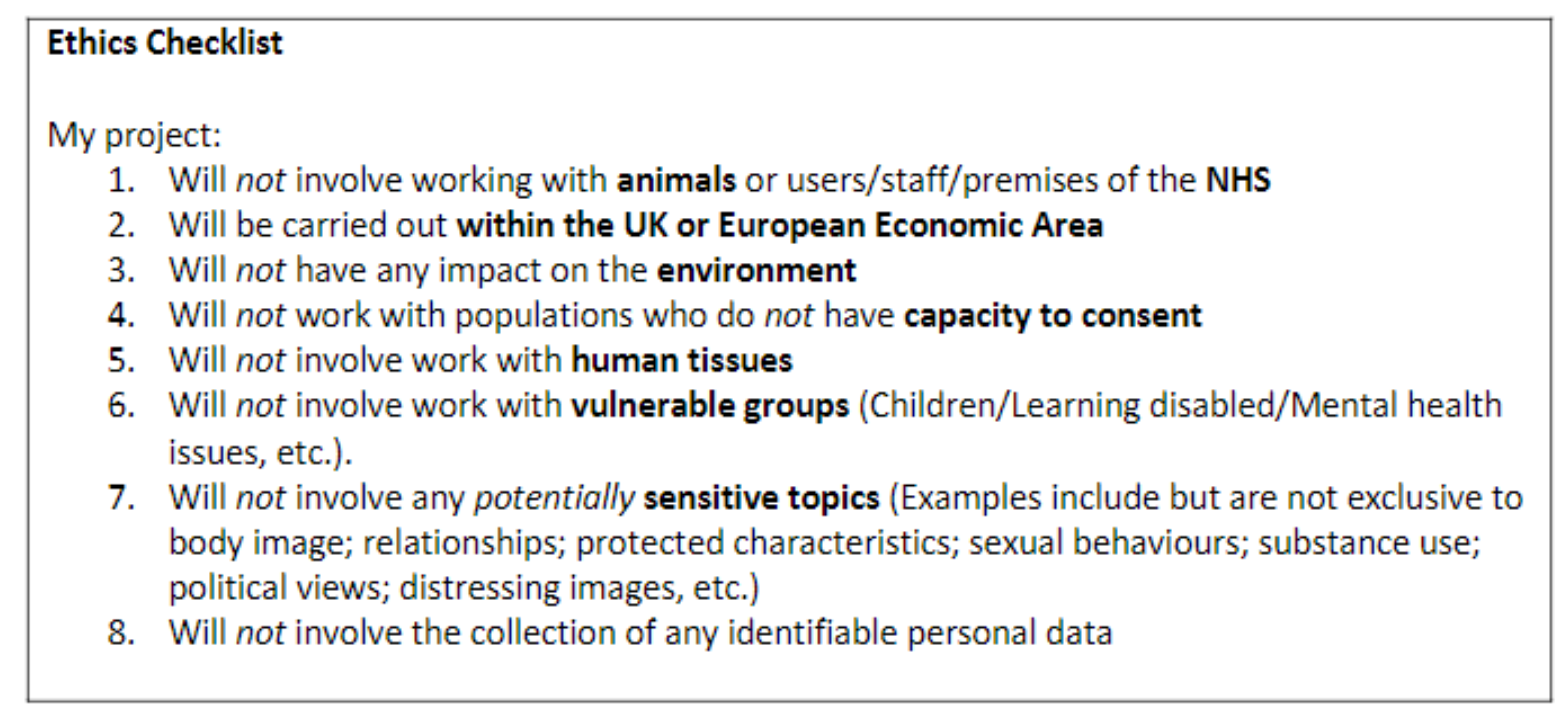
During the break, I aim to strike a balance between rest and work, focusing on optimizations, result collection, and completing a first draft of my dissertation by the break's end. Dissertation progress will continue gradually, with a concentrated effort during the Easter break to complete a final draft. I plan to start collecting results slightly before the break to ensure I have enough time for analysis. The project plan allows for flexibility in time allocation between implementation stages and the break. Project tweaks will adapt based on available time, potentially extending past the break while I am waiting for dissertation feedback from my supervisor.

The main project risk is not completing the implementation by the March 22nd presentation deadline, before the Easter break. This may impact achieving objectives, particularly implementing parallelized SPH algorithms for GPU architectures. This deadline has specifically been chosen to mitigate the impact of not completing the implementation by the deadline as it allows for flexibility during the break to catch up on any outstanding work, at the expense of less time to rest during the break. Additionally, I will be careful in how and what I choose to implement from the SPH literature to ensure that I am balancing optimization of the project with time, allocating most of the time to optimize my work until after I have a completed working simulation.

Another risk is overemphasis on project optimization during the break, leading to worse dissertation work. To manage this, I'll focus solely on completing the first draft of the dissertation after collecting the initial results. From there, I will be able to prioritize optimizations while iteratively collecting new results to refine the results section of my dissertation draft.

Ethics:

Given my project focus on fluid dynamics and its objectives, there are no direct ethical considerations to address as there is no involvement of external parties.



References:

[1] Koschier, Dan & Bender, Jan & Solenthaler, Barbara & Teschner, Matthias. (2022). A Survey on SPH Methods in Computer Graphics. Computer Graphics Forum. 41. 737-760.

[2] Ihmsen, Markus & Orthmann, Jens & Solenthaler, Barbara & Kolb, Andreas & Teschner, Matthias. (2014). SPH Fluids in Computer Graphics - Eurographics State-of-the-art report.

[3] Teschner, Matthias & Heidelberger, Bruno & Müller, Matthias & Pomeranets, Danat & Gross, Markus. (2003). Optimized Spatial Hashing for Collision Detection of Deformable Objects. VMV’03: Proceedings of the Vision, Modeling, Visualization. 3

[4] Green, Simon. 2010. “Particle Simulation Using Cuda” NVIDIA Whitepaper 6:121-128

[5] LIU M., LIU G.: Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments. Archives of Computational Methods in Engineering 17, 1 (2010), 25–76.

[6] PRICE D. J.: Smoothed Particle Hydrodynamics and Magnetohydrodynamics. Journal of Computational Physics 231, 3 (Feb. 2012), 759–794.