Dumoulin Glenn

Go with the Flow: Synchronizing a GPU-based fluid simulation over a network

Supervisor: Vanden Abeele Alex

Coach: Geeroms Kasper

Graduation Work 2024-2025

Digital Arts and Entertainment

Howest.be

CONTENTS

ABSTRACT & KEY WORDS

An abstract explains the outline of the paper concisely (the methods, results, etc.). Maximum length of 250 words, preferably both in English and Dutch.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis. Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

PREFACE

A preface is a statement of the author's reasons for undertaking the work and may include personal comments that are not directly relevant to other sections of the thesis or dissertation. No word count limit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis. Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

LIST OF FIGURES

The list of figures lists the figures in the order in which they appear throughout the thesis. They may be numbered sequentially, or be subdivided following the chapters in which they appear.

INTRODUCTION

In the introduction, you write the background of your topic and discuss the observation that spurred you on to do this research project. Explain the purpose of the paper and present your research question(s) and the hypothesis at the end of this section. This section is typically a couple of pages long.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

LITERATURE STUDY / THEORETICAL FRAMEWORK

1. FLUID SIMULATIONS

Fluid simulation is a technique used to replicate and visualize the behaviour of fluids (liquids and gases) in a virtual environment. These simulations have a wide range of applications, from visual effects for movies and games to scientific research in fields such as aerodynamics and weather simulations. [1]

The primary goal of a fluid simulation is to represent the dynamics of fluid flow in a realistic manner. This is achieved by solving mathematical models that describe the movement of fluid particles and the forces acting upon them. Among these models, the Navier-Stokes equations play a pivotal role.

The Navier-Stokes equations are partial differential equations which describe the motion of viscous fluid substances. They lay the foundation to most fluid simulations and are derived from the principles of conservation of mass, momentum, and energy. [2] In summary this means that the equations ensure that mass is neither created nor destroyed within the fluid and account for forces like viscosity, pressure, and external forces (such as gravity).

Over the years, several methods have been developed to simulate fluid dynamics, with particle-based and grid-based simulations being the two primary approaches.

1.1. PARTICLE-BASED SIMULATIONS

In particle-based simulations the fluid is represented by individual particles that interact with each other and their environment. Every particle is a small portion of the fluid, and handles its own position and velocity as it moves through the fluid. One of the most common methods used for particle-based simulations is the Smoothed Particle Hydrodynamics (SPH) method.

In SPH, fluid properties, such as density, pressure, and velocity, are calculated based on the surrounding particles, creating a physically accurate approximation of fluid flow. SPH also guarantees conservation of mass without extra computation since the particles themselves represent mass. Particle-based methods are especially effective for simulating free-surface flows, such as water in a container, or for applications requiring realistic visual effects, like splashes, waves, and interactions with solid objects. [3], [4]

One drawback over grid-based techniques is the need for large numbers of particles to produce simulations of equivalent resolution. However, accuracy can be significantly higher with sophisticated grid-based techniques, since it is easier to enforce the incompressibility condition in these systems. SPH for fluid simulation is being used increasingly in real-time animation and games where accuracy is not as critical as interactivity. [3]

1.2. GRID-BASED SIMULATIONS

In grid-based simulations the fluid is divided into a fixed grid of cells, and each grid cell contains the fluid properties (such as velocity, pressure and density) at that point in the fluid simulation. The simulation is updated iteratively for each time step, updating the properties of the fluid for every grid cell.

These methods are typically more stable and efficient than particle-based simulations, particularly for large-scale simulations. One of the most widely used techniques for grid-based simulations is the "Stable Fluids" algorithm by Jos Stam. This approach is commonly used in both fluid dynamics research and computer graphics, especially for simulating fluids in constrained environments (such as rivers, oceans, or fluid flows within a confined space). [5], [6]

The “Stable Fluids” approach is based on the Navier-Stokes equations. These equations are notoriously hard to solve when strict physical accuracy is of prime importance. The “Stable Fluids” solvers on the other hand are geared towards visual quality. The emphasis is on stability and speed, which means that the simulations can be advanced with arbitrary time steps. [5], [6]

1.3. GPU ACCELERATION

Both particle-based and grid-based simulations can benefit significantly from GPU (Graphics Processing Unit) acceleration. GPUs are designed to handle large amounts of parallel computations, which makes them particularly well-suited for the computationally demanding tasks required by fluid simulations. [7] By offloading the bulk of the simulation calculations to the GPU, the simulation can run much faster, enabling real-time fluid dynamics in interactive applications such as video games, virtual environments, or simulations for scientific visualization.

In practice, grid-based fluid simulations are more commonly accelerated on the GPU because the regular structure of the grid lends itself well to parallelization. For instance, calculating the velocity and pressure fields for each grid point can be done in parallel, dramatically speeding up the simulation process. Particle-based methods can also be accelerated on the GPU, but the irregular structure of particles can make parallelization more challenging, requiring more complex algorithms for efficient execution. [8]

As fluid simulation techniques continue to evolve, the need for real-time, high-quality simulations in interactive environments will only increase, making GPU acceleration a critical aspect of fluid dynamics research and applications.

2. DETERMINISM IN GPU-BASED ENVIRONMENTS

This chapter will shortly explain determinism, apply it to floating-point arithmetic, and introduce a solution to guarantee deterministic calculations.

2.1. DETERMINISM

Determinism refers to the property a system in which no randomness is involved in the development of future states of the system. A deterministic model will thus always produce the same output from a given starting condition or initial state. [9] This concept is crucial for simulations, particularly in networked or distributed environments where synchronization across multiple systems is essential.

GPU-based environments, while highly performant, pose unique challenges to achieving determinism due to the non-deterministic nature of certain operations and the architecture of GPUs. [10]

2.2. NON-DETERMINISTIC FLOATING-POINT ARITHMETIC

Floating-point arithmetic, the default representation for real numbers, is inherently non-deterministic due to several factors. The most well-known is the presence of rounding errors, introducing minute discrepancies that accumulate over time. This factor is something that we can work around. However, there are other factors that are a lot more difficult to resolve. [11]

There is no standardized implementation for floating-point arithmetic. This means that different GPU manufacturers (such as NVIDIA and AMD) may implement floating-point standards differently. Even within the same manufacturer differences between GPU models can affect floating-point precision and execution order. [11]

In networked simulations, these discrepancies can lead to desynchronization, where clients diverge in their state representations over time. More on these topics follows in the upcoming chapters.

2.3. FIXED-POINT ARITHMETIC

To address the non-deterministic nature of floating-point computations, fixed-point arithmetic is often employed. In fixed-point arithmetic, numbers are represented with a fixed number of digits after the decimal point, ensuring consistent behaviour across platforms. [12], [13]

Using fixed-point numbers also has its disadvantages, primarily the limitations of range and precision in comparison to floating-point numbers. Fixed-point numbers represent a floating-point number, but they are stored as an integer. That means that if you want a wider range of values you will have to sacrifice precision and vice versa. [13]

In GPU-based fluid simulations, fixed-point arithmetic ensures deterministic outcomes regardless of hardware differences. This is because fixed-point numbers are essentially integers with limited range and precision, and integer arithmetic is guaranteed to have deterministic results within its range.

3. NETWORKING

Networking forms the backbone of multiplayer systems, enabling devices to exchange data and synchronize states over the internet or local connections. This section introduces key networking principles, explores protocols and architectures, discusses challenges in real-time communication, and highlights Unity’s networking tools.

It is important to note that all information in this chapter will use the book “Multiplayer Game Programming” by Madhav Sanjay and Glazer Josh as a reference. So to avoid having to reference it for every paragraph I will reference it here at the top of this chapter and any additional references will be placed at paragraph level. [14]

3.1. NETWORKING BASICS

Networking involves the exchange of data between devices, typically referred to as nodes. In multiplayer systems, these nodes include clients and servers. Clients are devices operated by players, they are responsible for input handling and rendering game visuals. Servers are central or distributed systems that manage the game state and enforce rules. Typically one or more clients connect to a server to play together, although sometimes there are no servers and one of the clients has to step up to take the role of server. These nodes acting as both client and server are known as hosts.

Communication between nodes occurs through packets, which are small bundles of data transmitted over the network. Each packet consists of a header and a payload. The header contains metadata such as the source and destination for the packet. The payload contains the actual data such as player actions and object states.

Data transmission relies on the Internet Protocol (IP), which routes packets between devices using unique IP addresses.

Sockets act as endpoints for sending and receiving data. They pair an IP address with a port number to create a communication channel. Multiplayer games typically use multiple sockets for simultaneous data streams (e.g., game state updates, chat messages).

3.2. TCP AND UDP

Data transmission in multiplayer games typically uses one of two transport protocols, TCP or UDP.

TCP (Transmission Control Protocol) establishes a connection-oriented session, ensuring reliable delivery, error checking, and packet ordering. It is well-suited for scenarios where accuracy matters more than speed, such as login systems, file transfers or chat messages. [15]

UDP (User Data Protocol) is connectionless and focuses on speed rather than reliability. It is commonly used for real-time gameplay, where occasional packet loss is acceptable to maintain responsiveness. [15]

3.3. NETWORK TOPOLOGIES

A network topology determines how the computers in a network are connected to each other. In the context of a game, the topology determines how the computers participating in the game will be organized in order to ensure all players can see an up-to-date version of the game state.

There are two main types of topologies used by games, client-server and peer-to-peer.

3.3.1. CLIENT-SERVER ARCHITECTURE

In a client-server topology, one game instance is designated the server, and all of the other game instances are designated as clients. Each client only ever communicates with the server, while the server is responsible for communicating with all of the clients. Advantages include simplified synchronization and security, but the server can become a bottleneck.

There is also a subclassification of types of servers. Some servers are dedicated, meaning they only run the game state and communicate with all of the clients. The dedicated server process is completely separate from any client processes running the game. This means that the dedicated server typically is headless and does not actually display any graphics. The alternative to a dedicated server is a listen server. In this configuration, the server is also an active participant in the game itself.

3.3.2. PEER-TO-PEER ARCHITECTURE

In a peer-to-peer topology, each individual participant is connected to every other participant, distributing computational load. While this reduces server dependency, it introduces challenges in maintaining synchronization and security.

The concept of authority is much more nebulous in a peer-to-peer game. A common approach in peer-to-peer games is to share all actions across every peer, and have every peer simulate these actions. This model is sometimes called an input sharing model.

Furthermore, it is important to ensure that the game state is consistent between all peers. This means that the game implementation needs to be fully deterministic. In other words, a given set of inputs must always result in the same outputs. A few important aspects of this include using checksums to verify consistency of the game state across peer and synchronizing random number generation across all peers.

3.4. NETWORKING CHALLENGES

Networked games live in a harsh environment, competing for bandwidth on aging networks, sending packets to servers and clients scattered throughout the world. This results in data loss and delay not typically experienced during development on a local network.

In this section I will discuss latency and packet loss. Although these topics will most likely not pose as an issue for this research project, it is at least good to know about them in case the project were to be extended to support Wide Area Networks.

3.4.1. LATENCY

The word latency has different meanings in different situations. In the context of computer games, it refers to the amount of time between an observable cause and its observable effect. It is a commonly held misconception that networking delay is the primary source for latency in gameplay. While packet exchange over the network is a significant source for latency, it is definitely not the only one. A couple of non-network latency sources include input sampling latency, render pipeline latency, and vsync.

To handle latency there are multiple strategies such as client-side prediction, client-side interpolation, and server-side rewind. More information on these strategies can be found on the following references. [16], [17], [18]

3.4.2. PACKET LOSS

The largest problem facing network game developers is packet loss. It’s one thing if a packet takes a long time to get where it’s going, but quite another if the packet never gets there at all. This is not an issue when sending TCP packets, because TCP guarantees packet delivery.

However, as discussed earlier UDP is commonly used for real-time gameplay. So often developers will decide to create their own custom reliability layer on top of UDP.

Another technique to help maintain synchronization is called state synchronization, where the current game state is updated at regular intervals from the server to all clients. Clients can then check if they are still running a synchronized version of the game state. If this is not the case, that client can correct its own game state by interpolating to the server’s game state, or instantly correct to the desired game state if too much desynchronization occurred. [19]

3.5. NETWORKING IN UNITY

Unity provides networking solutions like Netcode for GameObjects and Unity Transport Package. It offers many tools, including tools to manage low-level communication, synchronize game objects across clients, and implement RPCs (Remote Procedure Calls) to trigger actions remotely. [20]

Unity also allows developers to integrate third-party libraries or implement custom networking layers for more control. [20]

CASE STUDY

Alternatively, as opposed to research, you might have opted for a case study. Whichever you choose, you detail the elements of your experiment(s), the tests, objects you will test upon and subjects you will test with, the data gathering, data cleaning or feature extraction, measurements, … and you present the results obtained in an objective manner for each of the tests you conducted.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

1. INTRODUCTION

In the introduction, you write the background of your topic, explain the purpose of the paper more broadly, and explain the hypothesis, and the research question(s).

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

2. MODELLING

2.1. BLOCKOUT

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

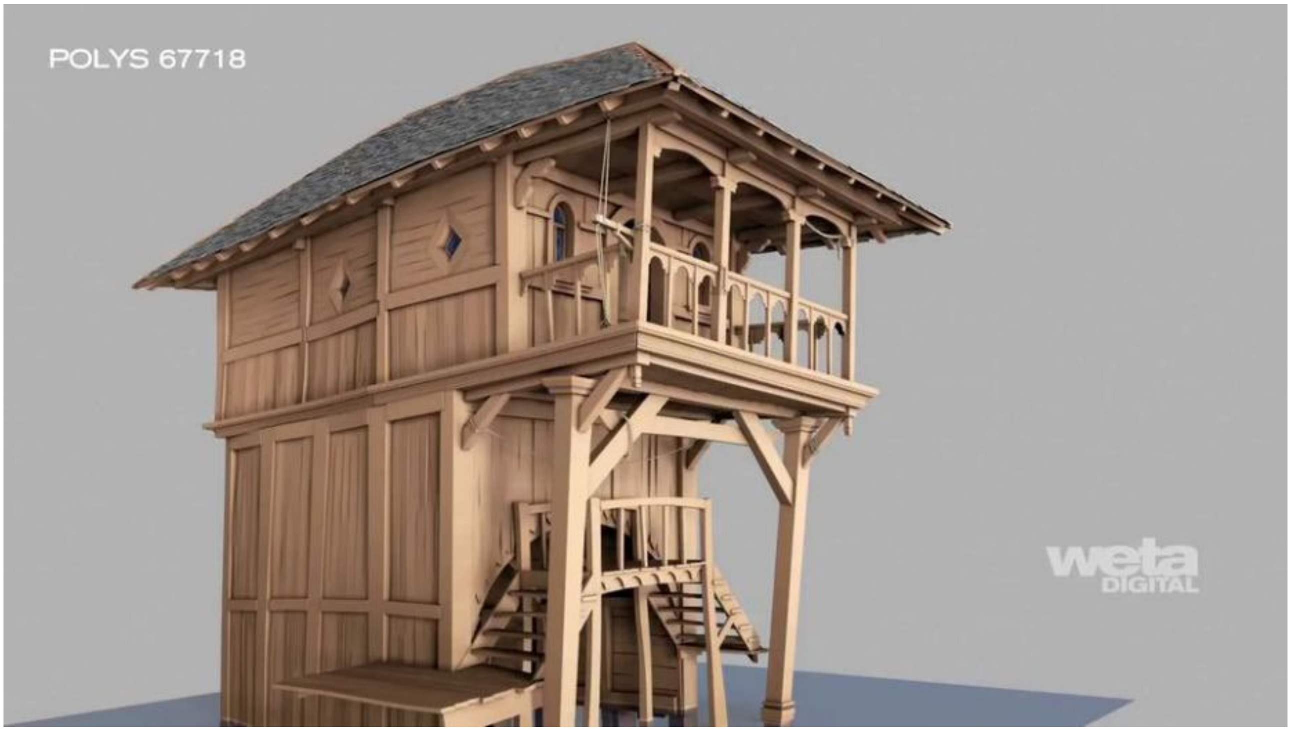

Figure 1 : MAKING OF THE HOBBIT: THE DESOLATION OF SMAUG – LAKETOWN (WETA DIGITAL, 2014)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

2.2. ZBRUSH

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

3. TEXTURING

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

4. SHADING

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

5. LIGHTING

L Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

DISCUSSION

In this section, you offer an interpretation of the results you obtained and try to relate them to the theoretical framework you presented. This is typically not a very long section, but obviously one of the most important ones.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

CONCLUSION

In this section, you ascertain the demonstrable outcomes of your study and outline the merits of the project for the academic field and the discourse community. This is typically not a very long section, but obviously also one of the more important ones.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

FUTURE WORK

A couple of ideas that could be further explored using this research project are listed below:

1. Expand the networking to support Wide Area Network connections. This will increase RTTs (Round-Trip Times) because the packets have to travel further. I don't expect there to be significant impacts on the performance and accuracy of the fluid simulation. I do think my approach of delaying inputs for synchronized execution will have to be changed because either some clients might not receive the inputs on time to be executed or the delay would be too long and become noticeable by users.

2. Expand the fluid simulation to 3D. This will increase the amount of data that needs to be sent over the network because vectors are now vector3's instead of vector2's. When the 3D simulation is functional, it would be a nice extra to add objects such as boats or barrels in the fluid and make them interact with the fluid.

3. Continuing on the 3D expansion, the fluid simulation used in my project only runs within a predetermined space. Researching how to apply this research for a fluid in an open-world environment could be an interesting challenge.

4. Add support for synchronizing the fluid simulation between computers and mobile devices. This can be very interesting and challenging to research because the fluid simulation heavily depends on the GPU to result in the best visuals the simulation can get. Most modern mobile devices do have some sort of GPU built in, but these are less powerful than those from computers.

CRITICAL REFLECTION

This section is typically associated with a bachelor paper, not other forms of serious writing. It allows the student to reflect on the learning outcomes, both academically and in terms of personal growth.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

REFERENCES

[1] "Computational fluid dynamics," *Wikipedia*. Aug. 19, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Computational_fluid_dynamics&oldid=1241121206

[2] "Navier–Stokes equations," *Wikipedia*. Oct. 19, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Navier%E2%80%93Stokes_equations&oldid=1251968674

[3] "Smoothed-particle hydrodynamics," *Wikipedia*. Jul. 19, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Smoothed-particle_hydrodynamics&oldid=1235431985

[4] J. J. Monaghan, "Smoothed particle hydrodynamics," *Rep. Prog. Phys.*, vol. 68, no. 8, p. 1703, 2005.

[5] J. Stam, "(PDF) Stable Fluids," *ResearchGate*, Oct. 2024, doi: 10.1145/311535.311548.

[6] J. Stam, "Real-Time Fluid Dynamics for Games," May 2003, [Online]. Available: https://www.researchgate.net/publication/2560062_Real-Time_Fluid_Dynamics_for_Games

[7] "Graphics processing unit," *Wikipedia*. Dec. 04, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Graphics_processing_unit&oldid=1261206237

[8] "Unity Fluid Simulation Tutorial: CPU & GPU Methods." [Online]. Available: https://daily.dev/blog/unity-fluid-simulation-tutorial-cpu-and-gpu-methods

[9] "Deterministic system," *Wikipedia*. Jun. 13, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Deterministic_system&oldid=1092966327

[10] T. B. Brown, "ML Tidbits: Nondeterminism on the GPU," Tom B Brown. [Online]. Available: https://medium.com/tom-b-brown/tensorflow-nondeterminism-on-the-gpu-a0e86125fd06

[11] "Floating Point Determinism," Gaffer On Games. [Online]. Available: https://gafferongames.com/post/floating_point_determinism/

[12] "Fixed-point arithmetic," *Wikipedia*. Dec. 16, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Fixed-point_arithmetic&oldid=1263443554

[13] "Decoding Numerical Representation: Floating-Point vs. Fixed-Point Arithmetic in Computing," DEV Community. [Online]. Available: https://dev.to/mochafreddo/decoding-numerical-representation-floating-point-vs-fixed-point-arithmetic-in-computing-3h46

[14] S. Madhav and J. Glazer, *Multiplayer Game Programming: Architecting Networked Games*, 1st edition. New York, New York: Addison-Wesley Professional, 2015.

[15] "TCP vs UDP: What's the Difference and Which Protocol Is Better?," TCP vs UDP: What's the Difference and Which Protocol Is Better? [Online]. Available: https://www.avast.com/c-tcp-vs-udp-difference

[16] "Improving performance with client-side interpolation | Unity Multiplayer." Accessed: Dec. 21, 2024. [Online]. Available: https://docs-multiplayer.unity3d.com/netcode/current/learn/clientside_interpolation/

[17] "Client-side prediction," *Wikipedia*. Oct. 10, 2021. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Client-side_prediction&oldid=1049191089

[18] M. Henning, *marcohenning/ue5-server-side-rewind*. (Oct. 12, 2024). C++. Accessed: Dec. 21, 2024. [Online]. Available: https://github.com/marcohenning/ue5-server-side-rewind

[19] L. Apperson, "Beginnging Game Development: Syncing Game State," Medium. Accessed: Dec. 21, 2024. [Online]. Available: https://medium.com/@lemapp09/beginnging-game-development-syncing-game-state-0d63dfe16019

[20] U. Technologies, "Unity - Manual: Multiplayer." [Online]. Available: https://docs.unity3d.com/6000.0/Documentation/Manual/multiplayer.html

ACKNOWLEDGEMENTS

In this section, you can thank people who contributed to your work in a meaningful way.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis. Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

APPENDICES

All the code that has been written to achieve this research project, handle the measurements and make the screenshots used in this paper has been made available online on GitHub. It also includes all results from measurements taken during the different stages of the project.

The repository is available at
https://github.com/GlennDumoulin/NetworkedFluidSimulation
and will be kept online as long as possible.