



**YAŞAR UNIVERSITY
FACULTY OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING**

**COMP4920 Senior Design Project II, Spring 2020
Advisor: Mehmet Ufuk Çağlayan**

**POF: Performance Optimized Fluid System
Final Report
(Bachelor of Science Thesis)**

By:

Baran Budak -15070001012

Cihanser Çalışkan -16070001020

İsmail Mekan -15070001048




Revision History

Revision	Date	Explanation
1.0	8.12.2019	Initial Final Report.
1.1	10.2.2020	- Abstract and Özet part renewed. - Revision history added.
1.2	17.2.2020	Warning page added.
1.3	2.3.2020	Introduction section writings updated.
1.4	7.3.2020	2. semester Gantt chart updated.
1.5	15.2.2020	Requirements section added.
1.6	5.3.2020	Class diagram and description is updated.
1.7	18.3.2020	Code parts and explanations updated in implementation section.
1.8	30.3.2020	Performance test and results added.
1.9	14.4.2020	- Cost analysis of second semester added. - Plagiarism statement updated.
2.0	27.4.2020	- Appendices of product manual added. - Table of contents updated. - List of figures updated. - List of tables updated.

PLAGIARISM STATEMENT

This report was written by the group members and in our own words, except for quotations from published and unpublished sources which are clearly indicated and acknowledged as such. We are conscious that the incorporation of material from other works or a paraphrase of such material without acknowledgement will be treated as plagiarism according to the University Regulations. The source of any picture, graph, map, or other illustration is also indicated, as is the source, published or unpublished, of any material not resulting from our own experimentation, observation or specimen collecting.

Project Group Members:

Name, Last name	Student Number	Signature	Date
Baran Budak	15070001012		24.05.20
Cihanser Çalışkan	16070001020		24.05.2020
İsmail Mekan	15070001048		24.05.20

Project Supervisors:

Name, Last name	Department	Signature	Date
Mehmet Ufuk Çağlayan	Computer Engineering		

ACKNOWLEDGEMENTS

We would like to thank Gizem Kayar for discussions and suggestions in the development of the project. This project is supported by Yasar University Computer Engineering Department.

KEYWORDS

Term	Description
Cell	Axis aligned bounding box is divided into small identical cubes.
Colour field quantity	It is a function that calculates how each particle is affected by all the other particles.
Gradient	The directional derivative of a scalar field gives a vector field directed towards where the increment is most, and its magnitude is equal to the greatest value of the change.
Grid	Series of vertical and horizontal lines that are used to subdivide AABB vertically and horizontally into cells in three-dimensional space.
Iso-surface	An isosurface is a 3D surface representation of points with equal values in a 3D data distribution which is the 3D equivalent of a contour line.
Marching Cubes	Marching cubes is a computer graphics algorithm, published in 1987 for extracting a polygonal mesh of an isosurface from a three-dimensional discrete scalar field.
NVIDIA Flex	NVIDIA Flex is a particle-based simulation technique for real-time visual effects created by NVIDIA company.
Polygonal Mesh	A polygon mesh is the collection of vertices, edges, and faces that make up a 3D object.
Unity 3D	Unity is a cross-platform game engine developed by Unity Technologies. Unity is used for developing video games and simulations for consoles and mobile devices.
Spatial Hashing	Spatial hashing is a technique in which objects in a 2D or 3D domain space are projected into a 1D hash table allowing for very fast queries on objects in the domain space.

Table 1: Keywords

ABSTRACT

POF system aims at providing more optimized and faster surface identification and visualization on particle-based fluid simulations.

This project is research-based. It is possible for the small parts of the structures can change during the project. We research possible solutions for the problem and examined a lot of research papers for the algorithms. We discussed the pros and cons of various methods and decided to use specific algorithms for the mentioned reasons.

The POF system divided into a structure that has various algorithms. A control panel (controller or handler) administers these algorithms that placed as substructures in the POF system.

We can list these substructures as Hash System, Surface Particle Recognizer and Visualization parts. Hash System is an imaginary structure and serves to search for data easily. Surface particle recognizer distinguishes the surface particles. Visualization part draws the surface particles vertices which is an implementation of the Marching Cubes algorithm [4].

ÖZET

POF sistemi, partikül bazlı sıvı simülasyonlarında daha optimize ve daha hızlı yüzey tanımlama ve görüntüleme sağlamayı amaçlamaktadır.

Bu proje araştırmaya dayalıdır. Proje sırasında yapıların küçük bölümlerinin değişmesi mümkündür. Sorun için olası çözümleri araştırdık ve algoritmalar için birçok araştırma makalesini inceledik. Çeşitli yöntemlerin artılarını ve eksilerini tartıştık ve belirtilen nedenlerden dolayı spesifik algoritmalar kullanmaya karar verdik.

POF sistemi, çeşitli algoritmalara sahip bir yapıya ayrılmıştır. Sistemin arayüzü olan bir kontrol paneli, (denetleyici veya işleyici) POF sistemindeki alt bileşenleri yönetir.

Bu alt yapıları karma sistemi (hash system), yüzey partikül tanıyıcı ve görselleştirme parçaları olarak listeleyebiliriz. Karma (hash) sistemi hayali bir yapıdır ve verileri kolayca aramaya yarar. Yüzey partikül tanıyıcı yüzey partiküllerini ayırt etmemize yarar. Görselleştirme bölümü, Yürüyen Küpler algoritmasının [4] bir uygulaması olan yüzey parçacıklarının köşelerini çizmemizi sağlar.

TABLE OF CONTENTS

PLAGIARISM STATEMENT	3
ACKNOWLEDGEMENTS	4
KEYWORDS	4
ABSTRACT	5
ÖZET	6
TABLE OF CONTENTS	7
LIST OF FIGURES	8
LIST OF TABLES	8
LIST OF ACRONYMS/ABBREVIATIONS	9
WARNING	10
1. INTRODUCTION	11
1.1. Description of the Problem	11
1.2. Project Goal	11
1.3. Project Output	11
1.4. Project Activities and Schedule	12
1.4.1 First Semester Schedule	12
1.4.2 Second Semester Schedule	13
2. REQUIREMENTS	14
3. DESIGN	15
3.1. High Level Design	15
3.1.1 Class Diagram	15
3.2. Detailed Design	16
3.3. Realistic Restrictions and Conditions in the Design	16
4. IMPLEMENTATION and TESTS	17
4.1. Implementation of the System	17
4.1.1. Problems and Solutions	17
4.1.2. Implementation of Hash Algorithm	17
4.1.3. Particle Neighbour Algorithm	19
4.2. Performance Tests and Results	20
4.3 Environment Testing	24
4.3.1. Availability of the Necessary Environment	24
5. CONCLUSIONS	25
5.1. Summary	25
5.2. Cost Analysis	25
5.2.1 Cost Analysis of First Semester	25
5.2.1.1 Cost of Workers	26
5.2.1.2 Cost of Software	26
5.2.1.3 Cost of Hardware	26
5.2.1.3.1 PC components that used in Project	26
5.2.1.3.2. Optimal Simulation Computer	27
5.2.2 Cost Analysis of Second Semester	27
5.2.2.1 Cost of Workers	27
5.3. Benefits of the Project	28
5.3.1 Animations and Movies	28
5.3.2 Scientific work	28
5.3.3 Games	28
5.3.4 Construction	28

5.4. Future Work	28
References	29
APPENDICES	30
APPENDIX A: REQUIREMENTS SPECIFICATIONS DOCUMENT	31
APPENDIX B: DESIGN SPECIFICATIONS DOCUMENT	47
APPENDIX C: PRODUCT MANUAL	66

LIST OF FIGURES

Figure 1: Class Diagram	15
Figure 2: Cell ID Numbering	18
Figure 3: Group Struct	18
Figure 4: Hash Size	18
Figure 5: Intersection Boundaries Check	19
Figure 6: Finding Corner Cells	20
Figure 7: Finding Dimensional Cell Count	20
Figure 8: Finding Cell Numbers in Specific Area	20
Figure 9: Particles in Perfect Cube Form	21
Figure 10: Particles on Ramp	22
Figure 11: Particles are Crashing to Obstacles	22
Figure 12: Surface Recognition Time Scale by Frame Chart	23
Figure 13: Surface Recognition by particle number comparison chart	24

LIST OF TABLES

Table 1: Keywords	4
Table 2: List of Acronyms/Abbreviations	9
Table 3: Requirements List	14
Table 4: Problems and Solutions	17
Table 5: Environment Availability	25
Table 6: 1. Semester Cost of Workers	26
Table 7: Cost of Software	26
Table 8: Cost of PC1	26
Table 9: Cost of PC2	27
Table 10: 2. Semester Cost of Workers	27

LIST OF ACRONYMS/ABBREVIATIONS

AABB	Axis Aligned Bounding Box. Bounding volume for a set of objects is a closed volume that completely contains the union of the objects in the set.
API	Application Programming Interface.
CPU	Central Processing Unit.
CUDA	Compute Unified Device Architecture. CUDA is a parallel computing platform and application programming interface (API) model created by NVIDIA.
D3D11	Direct3D 11. Is to create 3-D graphics for games and scientific and desktop applications.
GPU	Graphic Processing Unit.
MVC	Stands for Model View Controller. MVC is an application design model comprised of three interconnected parts (Model, View, Controller).
NVIDIA	NVIDIA corporation is a company designs GPUs.
OPENGL	Open Graphics Library is a cross-language, cross-platform application programming interface for rendering 2D and 3D vector graphics.
POF	Performance Optimized Fluid.
SSF	Screen Space Fluids Pro.

Table 3: List of acronyms/abbreviations

WARNING!

Important Note: POF project has hardware-based requirements. Your GPU must have CUDA 8.0.44 or better version and D3D11 support. If you do not have the required components, POF will not work.

We were using the Yaşar university computer lab in the first semester. Since Yaşar University is closed because of the COVID-19, we cannot access the computer laboratory. Therefore, we cannot make any progress in visualization.

The %75 of the project is finished. Implementation of the Marching Cubes algorithm which is the last step about the visualization part of our project could not be completed (We have a working marching cubes code as a prototype. However, we did not implement to the POF system.). For this reason, we have restated our project requirements and goals which will be clarified detailed in the Final Report and Requirements Specifications Document. In brief, the implementation and testing of the surface recognition system is the new goal of our project and some of the requirements are discarded such as Marching Cubes.

1. INTRODUCTION

This section explained in three main titles: problem description, project goal and project output.

1.1. Description of the Problem

The main problem of the particle-based fluid simulation system is excessive numbers of the particles. A particle is a rigid body sphere.

There are millions of particles in a small number of liquids. Simulation control particles by physics-based calculations to obtain fluid behaviours. Simulation having difficulties in calculations dependent on a surplus of particles and time and memory complexity increases indirectly. Visualizing millions of particles on a scene are a tedious job.

1.2. Project Goal

During the POF project, we were researching ways of enhancing the performance and efficiency of particle-based fluid simulation. Creating a suitable and stable platform for executing a particle-based fluid simulation is one of our primary project goals. This platform is Unity for our project. Constructing a more user-friendly platform for testing and comparing various algorithms for scientific research is another project goal.

We aim to achieve these goals by reaching particles faster by constructing a spatial hash sub-system. To detect surface particles, we must implement surface recognition sub-system.

Our project has no predetermined method because POF is a research and development based. We can research and implement new methods during the project.

1.3. Project Output

- Better performance.
- Better memory efficiency.
- Fluid-like appearance and behaviour.
- Different algorithms testing for performance and efficiency.
- Surface particles detection.