iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** XII **Month of publication:** December 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Review on Smoothed Particles Hydrodynamics Simulation and Rendering

Deepa S. Kadam[1], Ram B. Joshi[2]
[1,2]*Computer Engineering Department, Indira College of Engineering and Management, Savitribai Phule Pune University, Pune*

Abstract—Smoothed Particle Hydrodynamics (SPH), a computational method for simulating fluid flows is easily parallelizable, as the interaction between two particles is independent of the others. For high quality conception at very high frame rates, a new, proficient interpreting pipeline can be expended. It's a challenging task to parallelize the particles by using thread while carrying out SPH simulation on GPU. In order to attain real-time simulation, a significant amount of computation is needed to calculate the interaction between the particles. Although the computation is highly independent, the number of particles involved are huge. The state of each particle for each frame based on its interaction with all other particles need to be computed.
Keywords— Smoothed Particle Hydrodynamics, parallelization, thread, simulation, particle interaction

I. INTRODUCTION

Fluid, a substance that continually deforms or flows under an applied shear stress. All liquids and gases are categorized as fluids. Differentiating factor between them is generation of free surface and flow ability. Liquids form a free surface while gases do not. Fluid does not resist deformation property. It has the ability to flow according to shape. For generating realistic animations of fluid objects and behavior, fluid simulation is a popular tool in computer graphic fields. Fluid simulation basically involves physics and mathematical equation to simulate fluid objects. Simulation was initiated with the introduction of computational fluid dynamics (CFD). Generally, CFD uses numerical methods and algorithms to analyze and solve fluid modeling and simulation by means of computers.

Finite difference methods (FDM) and finite element methods (FEM)- the traditional grid-based numerical method have troubles in handling some complex phenomena. This was the inspiration for researchers to strive for options to solve the problems. The SPH method has become a very good choice. The reason to use SPH over other numerical methods such as Particle in Cell (PIC), FDM and FEM is that SPH does not depend on the boundary conditions. It means, SPH does not need a grid to calculate any spatial derivatives. Instead, it applies analytical differentiation of interpolation formulae. The equations of momentum and energy become sets of ordinary differential equations which are easy to understand in mechanical as well as thermo dynamical terms.

Smoothed Particle Hydrodynamics is a gridless numerical method that can be used to simulate fluid flows with one or more free surfaces. It is a computational manner used for simulating fluid flows. Actually, SPH is a Lagrangian approach (in which coordinates move with the fluid) for computational fluid dynamics (CFD). It is an interpolation method for particle system and is also known as a mesh-less or grid-less method. This method was formulated by Lucy (1977), Monaghan and Gingold (1977). SPH method was actually developed for use in astrophysics areas and was first tested as numerical solution for gas flow problems for astronomical interest. The institution of pressure and force coupled with detailed analysis has allowed SPH to solve a wide variety of astrophysical problems.

The Smoothed-particle hydrodynamics method works by dividing the fluid into a discrete element's set. In SPH, these discrete elements are referred to as particles. These particles have a spatial distance that is known as the smoothing length, over which their properties are smoothed by a kernel function. Means by summing the relevant properties of all the particles which lie within the range of the kernel or the block, the physical quantity of any particle can be obtained. Generally, in SPH simulation four basic phases need to be performed for each particle.

Phase 1: Finding all neighbors within the smoothing length H

Phase 2: Density Phase: Update the density and pressure for each particle based on these neighbours

Phase 3: Force Phase: using updated density and pressure value, update pressure and viscosity force, detect collision, apply external force, and calculate acceleration.(depends on result from density phase)

Phase 4: Physics Phase: Step the particle to new position and velocity within time step t. (depends on result from force phase)

After these phases, graphics processing unit (GPU) data is copied to CPU.

The section II describes study of various effective techniques used in SPH, search intention introduced. Finally, section III is a conclusion.

II. RELATED WORK

A. Smoothed Particle Hydrodynamics On GPUs

An animation is an exceedingly horsy topic for all involves huge computation of particles in real time simulation. To produce animations, physically based simulation is used extremely in computer graphics. What includes animations is a faster movement of particles means the faster simulation. In real time applications the computational burden of these particles is very high. Takahiro Harada [1] achieved accelerated smoothed particle hydrodynamics, a simulation method for free surface flow by using graphics processing units (GPUs). Traditional algorithms do not satisfy all the requirements which are important to carry out real-time simulations. A Smoothed Particle Hydrodynamics (SPH) method for incompressible fluid was implemented using CUDA on GPU by Harada. Neighboring particles have to be searched to compute a force on a particle. However, this implementation of a neighboring particle search on GPUs is not that much easy. A method that can search for neighboring particles on GPUs developed.

Four steps used for GPU based SPH simulation. Meaning that, each time step of this GPU SPH is made up of:

Bucket generation: passes are made for particles and for each pass the particle indices are stored in increasing order. For bucket generation color mask, depth buffer and stencil buffer are used in different way. Calculated indices written to bucket index texture.

Density Computation: the weighted sum of mass of neighboring particles, distance from wall is calculated. Calculated density written to density texture.

Velocity Update: for computing pressure and viscosity. For computing pressure distance functions are used and calculated velocity marked to velocity texture.

Position Update: Velocity texture provides position updates which further calculated with explicit Euler's integration.

This GPU method enabled to carry all the computations of SPH simulation entirely on GPUs. All of the computation is done on GPUs and no CPU processing is needed, the proposed algorithm can exploit the massive computational power of modern graphics hardware that is GPUs. Consequently, as GPUs are very good choice for dealing with simulations. The experiment results show that our GPU based SPH implementation can achieve the rate of 89 frames per second in the simulation of 102K particles and gain nearly 140× speedups compared with the serial algorithm. So the speed-up boost also becomes inseparable part of this high computational method for SPH.

B. Defining Point Set Surfaces

Because of improved technologies for capturing points from the surfaces of real objects and because improvements in graphics hardware now allow us to handle large numbers of primitives, modeling surfaces with clouds of points is becoming feasible. This is interesting, since constructing meshes and maintaining them through deformations requires a lot of computation. It is useful to be able to define a two-dimensional surface implied by a point cloud. Such point-set surfaces are used for interpolation, shading, meshing and so on [2]. David Levin's MLS surface [Levin 2003] used for modeling and rendering with point clouds, was originally defined algorithmically as the output of a particular meshless construction. It has proved to be a very useful example of a point-set surface. Levin defined the MLS surface as the stationary points of a map f , so that x belongs to the MLS surface exactly when f (x) = x. This definition is useful but it does not give much insight into the properties of the surface. Surfels are the points equipped with normal. A point-set surface which takes a set of surfels as a input rather than a point cloud. Normals are often available, and using surfels rather than points makes the surface more robust in the face of both undersampling and of irregularities in the distribution of samples.

C. Preliminary Methods And SPH Equations

The history of computer graphics fluid simulation originates from the famous Navier-Stoke equations that describe the dynamics of fluids. This created a field of research in computing area with a special interest in fluid analysis called Computational Fluid Dynamics (CFD). The CFD focuses on studying fluid objects by means of computer graphic techniques and approaches. Since those equations are known and computers can solve them numerically, a large number of methods have been proposed in the area of CFD (Andreas, 2008). The fundamentals of any CFD equation are the Navier-Stoke equation. Eq (1) illustrates the Navier-Stoke equations for fluid motion.

$$\frac{\partial \bar{u}}{\partial t} + \bar{u}.\nabla\bar{u} + \frac{1}{p}\nabla p = \bar{g} + \upsilon\nabla.\nabla\bar{u}$$

Eq 1. Navier-Stoke equations for fluid

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

The equation is called '*momentum equation'* which describes how fluid moves due to external forces. The variables *u, ρ, g, v* represent velocity of fluid, density, acceleration due to gravity and kinematics viscosity respectively. The equation also describes the '*incompressibility condition'*. Navier-Stokes equations are describing the motion of a uid at any point within a ow by a set of non-linear equations, Foster and Metaxas (1996). The Navier-Stokes formulation of fluid motion is based on a grid structure. The earliest attempt in applying computer graphics to solve the Navier-Stoke equations was done by Foster and Dimitri (1996) who described the solutions to simulate liquids. According to Foster and Dimitri, realism is provided through a finite difference approximation to the incompressible Navier-Stoke equations and are coupled with the Lagrangian equations. According to the research, even the simplest animation exhibits subtle realistic behavior not available in previous computer graphics for fluid simulation. This realistic behavior then expanded by Stam(1999). He applied the semi- Lagrangian advection and methods to solve the Navier-Stoke equations. This research creates a method to extract the actual free surface of the liquid. Other efforts by researchers (Chen et. al. 1997) solved the 2D Navier Stoke equations using a computational fluid dynamics method.

Stam and Chen mapped the surface onto 3D using pressures in fluid flow. This method achieved realistic real-time fluid surface behaviors by applying the physical laws of fluid and avoiding extensive 3D fluid dynamics computations. Their model allows multiple fluid sources to be placed interactively in a dynamic virtual environment. Since about two decades ago, a number of fluid simulation techniques have been developed by experts. The most commonly used to describe fluid flow are the Eulerian and Lagrangian techniques (Frank, 2003). The Eulerian method is also called grid method or a finite difference method while the Lagrangian approach is called a particle method (Brian, 1999). Following table shows the comparison of Lagrangian and Eulerian methods to describe fluid flow according to Liu (2003).

Table 1 refer to the comparison of Lagrangian and Euclerian methods that are essential while dealing with smoothed particles hydrodynamics and their equations. The comparison is based in distinct types associated with SPH

TABLE I
COMPARISON OF LAGRANGIAN AND EUCLERIAN METHODS

Type	Lagrangrian methods	Eulerian methods
Grid/ Lattice	It is attached on moving Substantial	It is Fixed in space
Track/Trail	Movement of any point on materials on materials	Mass, momentum, and energy flux across grid nodes mesh cell boundary
Time Interval	Easy to obtain time- history data at a point attached on materials	Difficult to obtain time- history data at a point attached on materials
Moving boundary and interface	Easy to track moving boundary as computes force distribution	Problematic to track
Rough geometry	Easy to signify model with fine accuracy	Difficult to model with good accuracy
Large deformation	Hard to handle and formation switch	Easy to handle as it is in fixed space

D. Simulation And Rendering

Kipfer (et. al.,2006), presented an interactive technique for physic-based simulation and realistic rendering of rivers using Smoothed Particle Hydrodynamics (SPH). A design and implementation of a grid-less data structure to efficiently determine particles in close proximity is described. It's also suitable in resolving particle collisions. An altered collision detection data structure that can speed up the sparse particle system using a simple linear list is proposed in it. The structure had minimal memory consumption and remained static at runtime. It also presented the 'carpet method' by building a surface representation that covered all the generated particles and supported by gravity as external forces. This allows efficient rendering. As it constructed an explicit surface representation that is well suited for rendering, according to it, the used method is faster than the Marching Cubes approach. This study used 'carpet based method' to generate ground surface. Specifically, the terrain is created by increasing the maximum height

Volume 3 Issue XII, December 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

value for each node. It provides information for ground surface generation for the research.

E. Finding Neighbors Within Smoothening Length

As smoothed particle hydrodynamics (SPH) is based on interactions with the closer neighboring particles, implementing the neighbor list is a key point in terms of the high performance of the code. The efficiency of the method depends directly on how to build and use the neighbor list. In searching neighbours present in smoothening length the available searching algorithms for SPH codes are analyzed. Different gridding algorithms are evaluated, the gains in efficiency obtained from reordering of particles is investigated and the cell-linked list and Verlet list methods are studied to create the neighbor list.
Many problems in astrophysics span a wide range of densities. SPH performance is not so good with uniform smoothening length if the density have considerable dynamic range. SPH integrates the dynamical equations of motion for each fluid particles which represent points of the domain where physical properties are solved. These physical quantities are computed as an interpolation of the values of the closest neighbouring particles and are updated for each time step along the numerical simulation. The solution is to give each particle its own smoothening length, which can be determined at each time step. This particles smoothening length can be acquired by demanding that each particle interact with approximately the same number of neighbours (this provides the range). The dynamical equations in fluid dynamics represented by partial differential equations are moved to integral equations, which in SPH notation are transformed in sums over neighbouring points. A kernel function is used to determine the influence domain where each particle interacts with its neighbours. Thus, each particle is only influenced by the particles in its close neighbourhood, in such a way that only particles at a distance shorter than nh (n depends on the kernel choice and h is the smoothing length) should be evaluated [4]. For the sake of clarity a cut-off limit of 2h is considered. Thus, the determination of which particles are inside the interaction range requires the computation of all pair-wise distances, a procedure with high requirements in terms of computational time for large domains. The efficiency of this procedure, which involves a number of interactions on the order of N^2, is so poor that this brute force evaluation of interactions can only be used in academic exercises as pointed out in [5].

F. Efficient Neighbour Search For Particle-Based Fluids

Cell Indexing, a new approach for searching approximate neighbor particles necessary for efficient fluid simulation using SPH is used in [6]. Due to its inherent mesh-less nature Lagrangian particle-based animation- widely used strategy for simulating complex phenomena as fluids, the set of neighbor particles within a specified range must be efficiently found. Cell indexing approach encoded their coordinates and index into a key instead of storing particles into a fixed 3D grid or hash map. The list of keys is then sorted using linear time radix sort. A simple traversal using H –mask which can substantially increase the precision of Spatial Hashing or 3D grids is executed which quickly accumulate approximate neighbors without problematic cache misses of Spatial Hashing, large memory requirements of full 3D grids or O (n log n) time complexity of kd-trees. Sub-cell precision 1 by using larger H -masks, while having only constant factor slowdown attained.

III. CONCLUSIONS

The techniques attained designate numerous simulations suitable for real-time simulation as well as rendering of particles in fluids. The particles simulation methods which are essential for particles computations. The real-time fluid simulation on the GPU can be performed and gained, but fine-tuning it to be as optimized as possible proved to be quite challenge. Smoothed Particle Hydrodynamics, SPH, is a method developed as an attempt to model continuum physics avoiding the limitations of finite difference methods. It has been used in a wide variety of astrophysical applications and hydrodynamic problems. In coastal engineering, the problems are associated with propagating waves across the nearshore region, through the breaker line, and up the beach face. The SPH method is capable of dealing with problems with free surface, deformable boundary, moving interface, especially wave propagation and solid simulation.The particles sorting and rendering is methods need to be applied. A techniques to track particles in fluids and simulating them. SPH simulation is at interactive speed as this study discovered, the SPH method requires quite much computation to provide the sort of speed necessary for an interactive simulation. This can be useful in fluid dynamics.

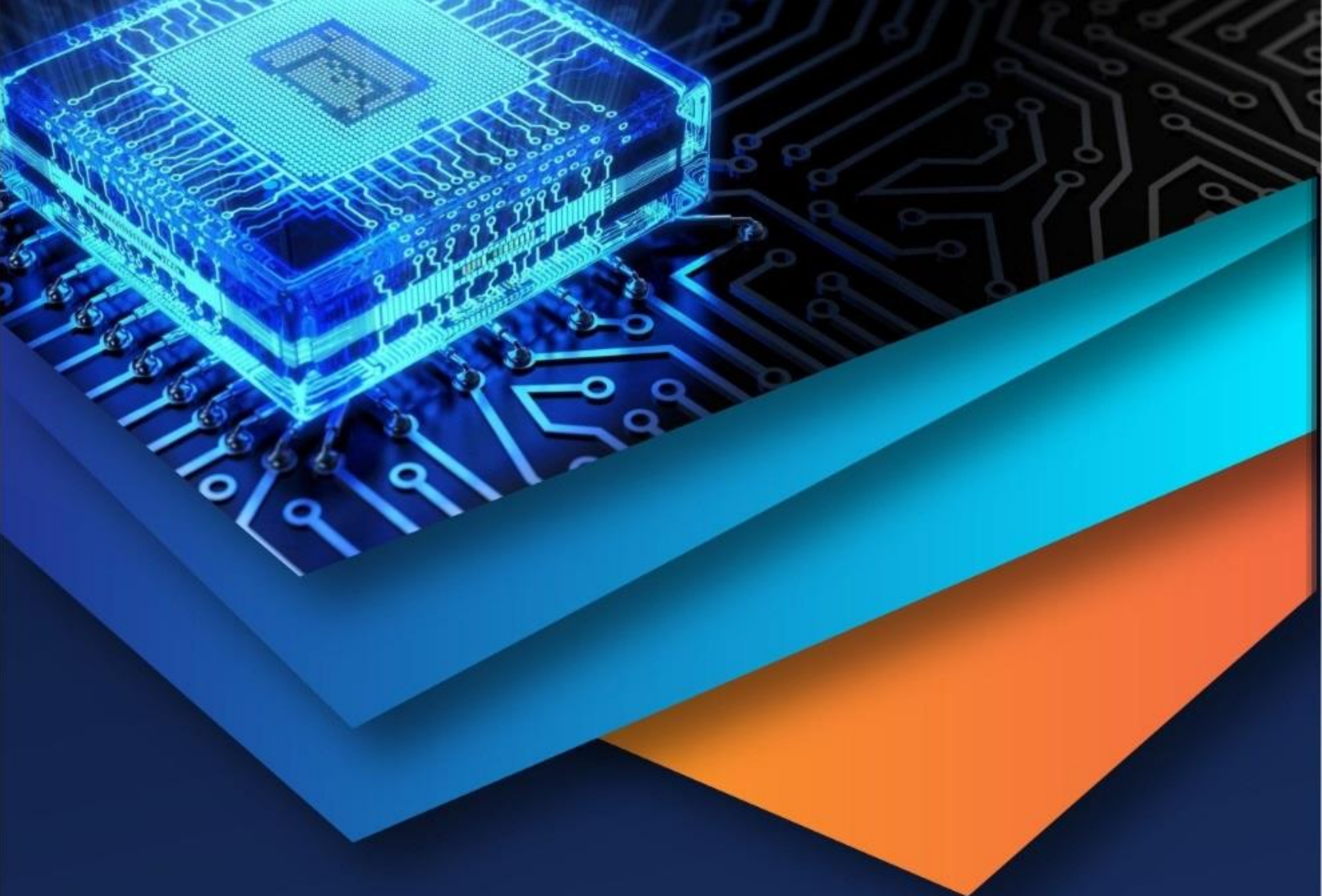
IV. ACKNOWLEDGMENT

The authors would like to thank Indira Group of Institutions and the Director/Principal Prof. Sunil Ingole, colleagues from Computer Engineering and Indira College of Engineering and Management, Pune Dist. Pune Maharashtra, India, for their support, counsels and inspiration.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

REFERENCES

[1] Harada T., Koshizuka S., Kawaguchi Y.: Smoothed particle hydrodynamics on GPUs. In Proceedings Computer Graphics International (2007), pp. 63–70.
[2] Amenta N., Kil Y. J.: Defining Point-Set Surfaces.Acm Transactions On Graphics 23, 3 (2004), 264–270.
[3] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., And Silva, C. T. 2003. Computing And Rendering Point Set Surfaces. Ieee Transactions On Visualization And Computer Graphics 9, 1, 3–15. An Earlier Version Appeared In Ieee Visualization 2001.
[4] Jm Dominguez, Ajc Crespo, M Gómez-Gesteira, Jc Marongiu: Neighbour Lists In Smoothed Particle Hydrodynamics. International Journal For Numerical Methods In Fluids 67 (12), 2026-2042
[5] Viccione G, Bovolin V, Carratelli Ep. Defining And Optimizing Algorithms For Neighbouring Particle Identification In Sph Fluid Simulations. International Journal For Numerical Methods In Fluids 2008; 58:625–638.
[6] Juraj Onderik, Roman Durikovic.: Efficient Neighbor Search For Particle-Based Fluids. Journal Of The Applied Mathematics, Statistics And Informatics (Jamsi), 2 (2007), No. 3
[7] Morris J. P.: Simulating surface tension with smoothed particle hydrodynamics. International Journal of Numerical Methods in Fluids 33 (2000), 333– 353
[8] Harada T., Koshizuka S., Kawaguchi Y.:Smoothed particle hydrodynamics on GPUs. In Proceedings Computer Graphics International (2007), pp. 63–70.
[9] NVIDIA: Research Zone https://research.nvidia.com/



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY