Final CFD Project Report by Group J

(Ashish Darekar, Jay Parmar, Sagar Garg)

Smoothed particle Hydrodynamics (SPH)

1. Introduction

Smoothed Particle Hydrodynamics is a meshfree, lagrangian particle method that originated from astrophysics and is today used in many fields like solid mechanics and fluid dynamics.
Instead of the traditional approaches of Eulerian meshes, employing numerical schemes like Finite Difference, Finite Volume and Finite Elements for solution of PDEs and integrals, the system is represented as a set of particles, each particle carrying information about its properties. For fluids, these might include density, mass, position, velocity, etc.

2. Motivation

The advantages become clear in cases where **geometries are complex** and often **time-consuming remeshing** is required. Problems of free surfaces, mobile boundaries, rapidly changing and complex topography such as breaking dam or turbulent flows are especially suited for a particle based approach. By SPH, these solutions are **numerically stable** without oscillations which are likely in a grid-based approach. Also, **mass is always conserved** since particles are treated individually. Finally, since each Lagrangian particle is treated individually, **evolution of the system in time is easily tracked graphically**, making our understanding more intuitive.

3. Method

The essence of the method is an interpolation scheme where only particles in a **local neighbourhood** of a particle A determine the forces acting on it. The local neighbourhood is determined by an 'influence of domain', of radius 'kh', outside which no particle exerts force.
The contribution of particles within the domain is determined by a **kernel function**- the nearer the particles, the higher the influence, in a gaussian manner. Among many different kernel functions present in literature, we choose the simple *cubic spline kernel.*

$$W = \text{factor} * \begin{cases} (\tfrac{2}{3} - q^2 - \tfrac{1}{2} q^3) & \text{for } 0 <= q < 1 \\ \tfrac{1}{6}(2-q)^3 & \text{for } 1 <= q < 2 \\ 0 & \text{otherwise}, \quad \text{factor} = (15/7)*pi*h^2 \end{cases} \qquad (1)$$

Any scalar property A is represented at position r by a weighted sum of contributions from particles in its influence domain, as per the kernel.

$A_a = m_b*(A_a/Rho_b)*W(\mathbf{X}_a-\mathbf{X}_b,h)$

The gradient of A, laplacian of A and divergence of any vector **A** can similarly be defined.

Fluid flows follow the equations of conservation of mass, energy and momentum balance. For our system, the following discretized equations are used:

Mass Conservation:

$$\frac{\partial Rho}{\partial t} = \Sigma m_b\, v_{ab}\, \text{gradW}(X_a\text{-}X_b, h) \qquad (2)$$

Momentum balance:

$$\frac{\partial v}{\partial t} = \Sigma m_b\, (P_a/Rho_a^2 + P_b/Rho_b^2)\, \text{gradW}(X_a\text{-}X_b, h) \qquad (3)$$

$$+2\nu_a \Sigma(m_b/Rho_b)*v_{ab}*(X_a\text{-}X_b)/|X_a\text{-}X_b|^2 * \text{gradW}(Xa\text{-}Xb,h) + g$$

For finding dynamic pressure, we approximate our system as a quasi-compressible flow and follow the Tait equation:

$$P_{dyn(a)} = B\,(Rho_a/Rho_0)^{\gamma} - 1) \qquad (4)$$

Where γ=7 and B (=85000) is a term related to density fluctuations of fluid.

4. Code Implementation

We have implemented two problems here:

1. Heat diffusion in a flat plate
2. Fluid flow in lid-driven cavity

Heat Diffusion in a flat plate:

We consider a solid plate with an initial distribution of equally-spaced 400 inner particles and 80 boundary particles and initial temperature set to 0°C. The bottom surface is then heated to 100°C and the other three maintained at 0°C, following Dirichlet boundary conditions.

The general algorithm is

Read the problem parameters
Set initial coordinates of particles
Set initial velocity, density, mass for inner particles

while(converge or maximum iteration not reached)
    Search for neighbouring particles
    Run the kernel. Update weights, gradients.
    Set temperature for boundary particles
    Calculate Laplacian of temperature
    CalculateTemperature
    Check convergence
    Print visualization at specific intervals

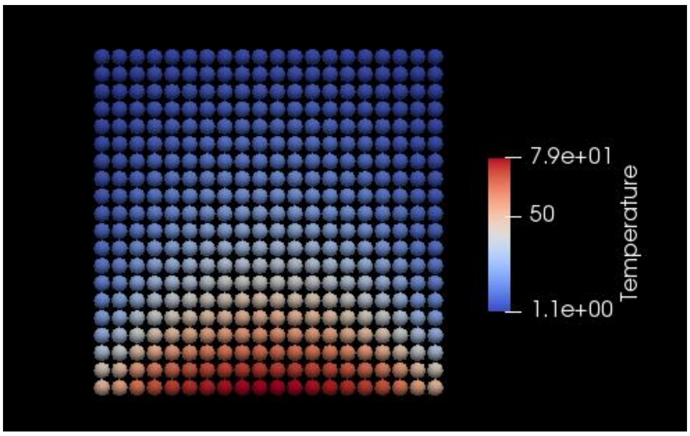
The major equation that has to be followed here is SPH version of the conservation of energy:

$$(\partial^2 T_a/\partial x^2 + \partial^2 T_b/\partial y^2)^m = 2*\Sigma\, m_b/Rho_b[\, T(x_a)^m - T(x_b)^m]\ \partial W/\partial r * (1/r_{ab}) \qquad (5)$$

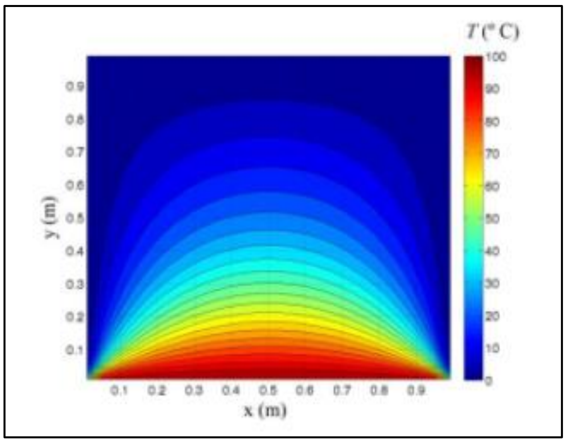
where m is the current time step.

The simulation ran till convergence (indication of steady state) and the results were compared to analytical solution. Good agreement was observed.

Paraview Results:



Results from Paper:



(Results from conference paper: CILAMCE 2016 Proceedings of the XXXVII Iberian Latin-American Congress on Computational Methods in Engineering Suzana Moreira Ávila (Editor), ABMEC, Brasília, DF, Brazil, November 6-9, 201)

The major error is at the boundaries where the kernel function does not sum up to unity, because there are fewer particles in the influence domain near the boundaries. This can be improved using a Corrected SPH method, which helps to normalize the kernel function, but has not been implemented here.

Fluid Flow in Lid-Driven Cavity

We now take on the more challenging problem of moving particles with the lid-driven cavity example. A hollow space walled on east, west and south borders with water flowing past at constant velocity (U=0.001m/s) from the top. The system consists of 400 inner moving particles and 160 fixed boundary particles (We doubled the boundary particles so as to enforce boundary conditions more strongly).

Algorithm:

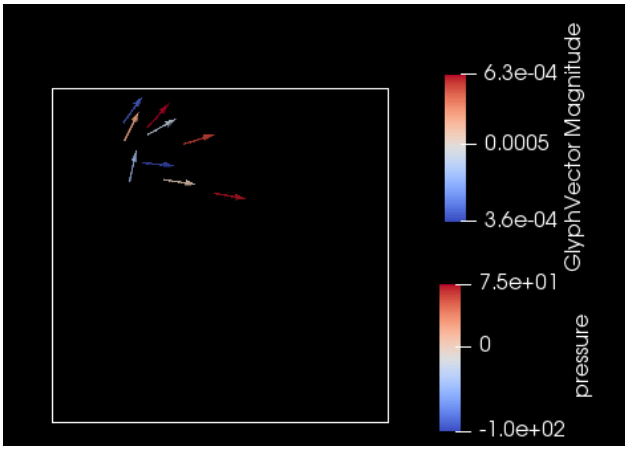
Read the problem parameters
Set initial coordinates of particles
Set initial velocity, density, mass for inner particles
while(converge or maximum iteration not reached)
    Search for neighbouring particles
    Run the kernel. Update weights, gradients.
    Set velocity, density, mass for boundary particles
    Calculate density as per Eq 2
    Update pressure as per Eq 4
    Calculate velocity as per Eq 3
    Calculate position.
    Check convergence
    Update position, velocity basis reflect boundary condition
    Print visualization at specific intervals

Boundary Condition:

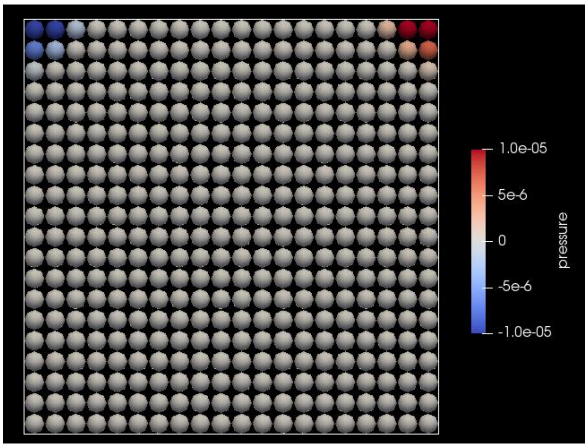
We apply dynamic boundary conditions on the three walls, such that a particle hitting the walls, reflects back with the same magnitude of velocity, but changes direction as if in an elastic collision. Other possibilities of damped collisions are also a possible option. On the top, boundary particles are fixed but have a velocity U=0.001 m/s horizontally, which is imparted to the inner particles as they come in the influence of domain. These are implemented in the reflect() function.

Paraview Results:

1. Cavity problem using SPH (with 9 particles) Cavity problem using SPH (with all particles)

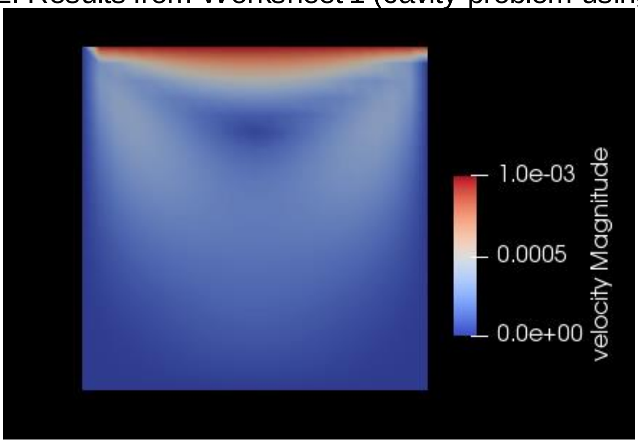


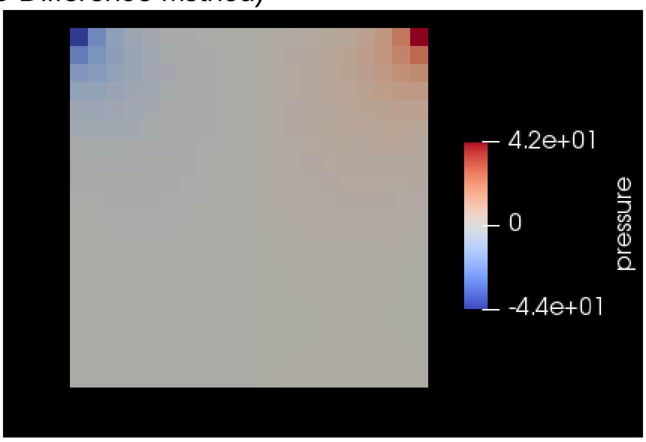
Velocity



Pressure

2. Results from Worksheet 1 (cavity problem using Finite Difference Method)





5. Conclusions:

SPH method is a powerful alternative to traditional mesh-based methods, and is especially in cases of complex geometries and fast-changing domain sizes. It is simpler to implement, more intuitive to visualize spatio-temporal evolution and does not have the numerical instabilities of many mesh-based schemes. Possible improvements include using a Corrected kernel function that normalizes near the boundaries as well. Currently, the code gives low accuracy on multi-dimensional flows, because of the kernel interpolation near boundaries where there aren't enough particles in the influence domain, hence truncating the shape of the kernel. This also makes the convergence slow. Also, the implementation of the boundary condition is not as natural as in other methods. There is a reliance on including ghost particles, which enforce the boundary conditions, but only softly. The other obvious improvement is the parallelization of the code, which would largely improve the performance. All together, SPH is a great technique with applications in numerous fields including astrophysics, atmospheric dynamics, and of course fluid flows.

6. References:

[1] Fraga Filho, C.A.D.: Development of a computational instrument using a Lagrangian particle method for physics teaching in the areas of fluid dynamics and transport phenomena. Rev. Bras. Ensino Fís. (online) 39(4), e4401 (2017). https://doi.org/10.1590/1806-9126-rbef-2016-0289

[2] Louis Goffin, Development of a didactic SPH model, 2012-13

[3] C. S. Nor Azwadi, Z. Mohd Shukri, M. Y. Afiq Witri, Numerical Investigation of 2D Lid Driven Cavity using Smoothed Particle Hydrodynamics (SPH) Method, The 4th International Meeting of Advances in Thermofluids, Melaka, Malaysia - 3rd & 4th October, 2011