Supervised Learning Project - 2020

Supervisor:

Prof. Prabhu Ramachandran
Department of Aerospace Engineering,
Indian Institute of Technology, Bombay,
Powai, Mumbai - 40076

Author:

KT Prajwal Prathiksh
Department of Aerospace Engineering,
Indian Institute of Technology, Bombay,
Powai, Mumbai - 40076

Abstract

This project is aimed primarily at replicating the research of Sun, Peng-Nan, et al. on the self-propulsive fish-like swimming hydrodynamics [1]

The work done by Sun is based on $\delta+$ - SPH scheme, which involves the δ - SPH framework along with a particle shifting technique (PST), and adaptive particle refinement (APR) which is a numerical technique adopted to refine the particle resolution in the local region and de-refine particles outside that region. These modifications provide the newly developed $\delta+$ - SPH scheme with higher numerical accuracy and efficiency. The fish is modelled using a NACA0012 aerofoil that performs periodic oscillations, similar to that of a fish in water.

Acknowledgements

I would like to express my gratitude to Prabhu Ramachandran, Professor, Department of Aerospace Engineering, IIT Bombay, for giving me the opportunity to work on this project, and for providing the motivation and resources which were crucial to the project. This project has been an invaluable learning exercise, which I am thankful for.

I wish to thank Abhinav Muta, Pawan Negi, and Amal S Sebastian. They have helped me in approaching the problem statement at hand, and discussions with them made it possible for me to obtain further clarity on the topic. Their inputs and suggestions were vital to this project.
I would like to thank IIT Bombay, for providing the freedom and a rich platform for students to explore and learn, well beyond the mandated academic curriculum.

Finally, I wish to acknowledge the continuous support of my family which I am indebted to.

Contents

List of Figures

1 Introduction

Smoothed particle hydrodynamics (SPH) is a technique for problem-solving in *Computational Continuum Dynamics* (CCD). This technique approximates numerical solutions of the equations of fluid dynamics by replacing the fluid with a set of particles. The equations of motion and properties of these particles are determined from the continuum equations of fluid dynamics by interpolation from the particles. The interpolant can be constructed using analytical functions, and spatial derivatives of the interpolated quantities can then be found using ordinary calculus. There is no need to use a grid, and the description of free surfaces, however complicated, is trivial.

This *Lagrangian* based particle formulation, therefore uses no background spatial mesh. Since there is no mesh to distort, the method can handle large deformations in a pure Lagrangian frame. Thus, material interfaces can be modelled naturally, and complex constitutive behaviour can be implemented accurately in a simple fashion. This allows for SPH to have for diverse and fascinating applications in various domains, that extend beyond the astrophysical and cosmological problems which it was originally designed to tackle.

This report is concerned with a problem involved in fluid mechanics, that occurs when the fluid interacts with a moving body. SPH has been used for many configurations involving moving bodies and fluids [2]. The primary aim was to simulate the propulsive motion of a fish-like swimming foil in a weakly compressible, viscous fluid. The motive beyond exploring this problem was to probe and explore the optimum hydrodynamic performance of the motion of a fish underwater. A man-made underwater vehicle that propels itself on a screw-propeller based system suffers from drawbacks such as the hydro-acoustic noise and cavitation erosion. Understanding the bionic (fish-like swimming) propulsion, that is more efficient, can produce better underwater vehicles, with a superior stealth property as well [1].

This domain which deals with the numerical simulation of bionic propulsion, compared to the observational work done that has been performed, is still in its infancy [3]. Attempts to model the complete motion of the fish and the fluid have not been so successful. The common practice involves the use of observational data from the motion of the fish to set-up the boundary condition on the flow, or analyse the vorticity distribution and the flow generated due to the swimmer. However, it still remains that, both the analytical and numerical simulations done so far, have only concentrated on the forward motion rather than attempt to model the more complex turns and rapid accelerations of a real fish. This domain, therefore, presents a plethora of unanswered questions that are yet to be solved.

2 Problem Statement

The primary goal of the project was to reproduce the results of Sun, Peng-Nan, et al. [1] on the self-propulsive fish-like swimming hydrodynamics, and subsequently build upon their research.

The work involved two distinct approaches to understand and investigate the propulsive nature of a swimmer.

1. **Approach 1**

   Similar to traditional approaches, this involves placing a fixed swimmer, that is placed in a uniform incoming free stream with a fixed inflow velocity of U - Fig 1. The hydrodynamic forces on the swimming body are measured, and when the stream-wise forces become negative, the fish is considered to be able to swim forward with a speed larger than the inflow velocity of U. Thus the swimmer is constrained to a fixed location by means of a varying external flow. Hydrodynamically, however, the state of a flapping swimmer fixed in a moving stream is different from that of the case involving a swimmer moving freely in a still fluid.

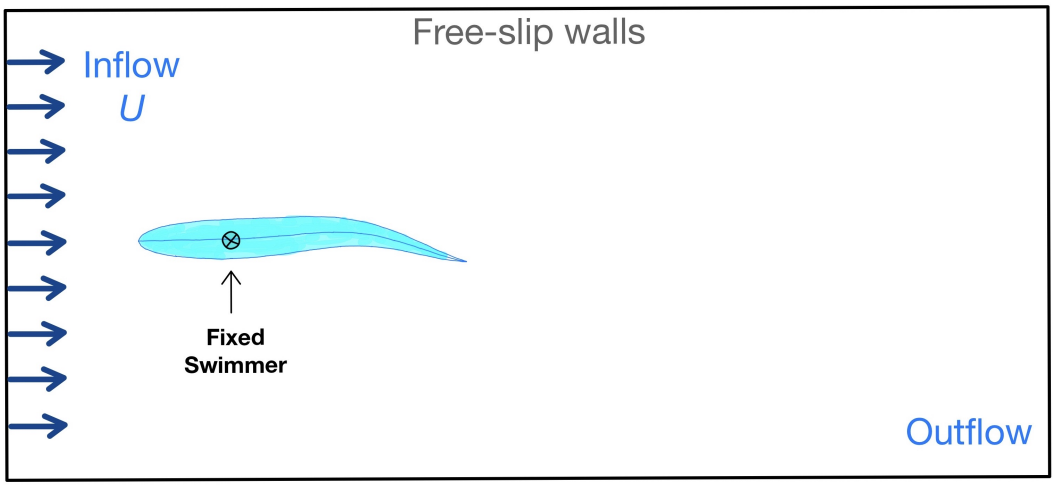


Figure 1: Approach 1

When simulating the swimmer in the approach, care needs to be taken while setting up the fluid domain so as to avoid blockage effects from the lateral walls which have a free-slip boundary condition. The effects of the out-flow boundary are also to be avoided. A negative drag coefficient would imply that the swimmer is capable of swimming forward with a velocity greater than U.

2. **Approach 2**

   This approach tries to mimic the natural process, with a free swimmer in a still fluid, where the swimmer is under the influence of the propulsive forces, drag forces, and inertial forces, to reach a dynamic equilibrium. Therefore the swimmer would move forward under a thrust force, which can be predicted by solving for the equations of motion governing the swimmer - Fig 2.

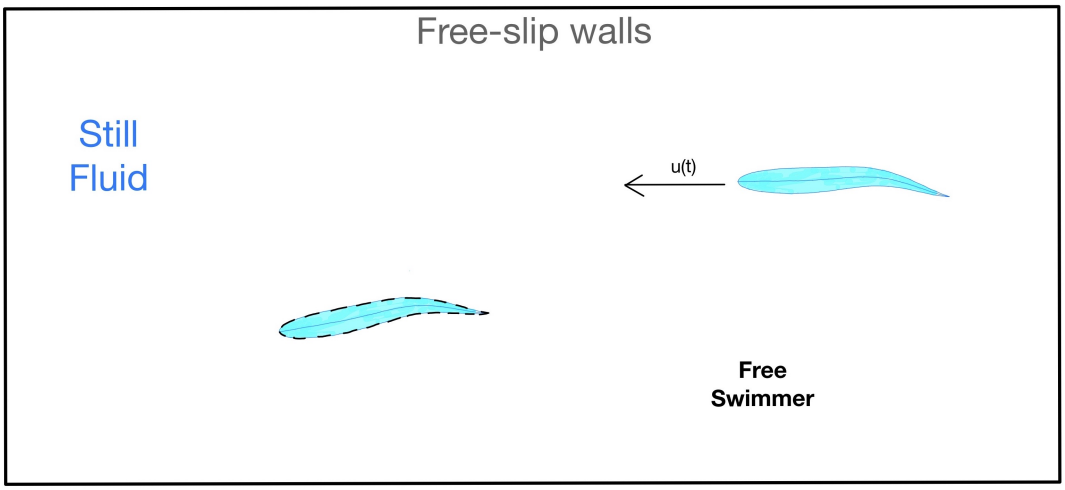


Figure 2: Approach 2

   This approach is observed to give the terminal swimming velocity with relative ease, as opposed to the former approach, which required several numerical trials to adjust the inflow velocity U and ensure the drag forces to oscillate around zero.

Therefore, the authors decided to use the second approach, for further analysis of the propulsive performance under different flapping conditions of the swimmer. The focus was on the forward propulsive speed, with the transversal and rotational motions of the swimmer being ignored.

Simulations were performed to analyse the dependence of the thrust force generated to the phase speed (c_p), as well as the amplitude of the oscillation on the tail of the swimmer. Vorticity field (an Eulerian property), as well as the Finite-time Lyapunov exponent(FTLE) [4] which are used to reveal the Lagrangian coherent structures (LCSs) (a Lagrangian property), were used to understand the flow physics in each case.

The terminal velocity was found to increase with increasing phase speed (c_p). The terminal velocity also seemed to reach a maximum with respect to increasing flapping amplitude, thereafter the terminal velocity decreased with increase in the amplitude.

2.1 SPH Scheme

The authors employed the $\delta+$ - SPH Scheme to solve the problem at hand. This particular scheme was chosen, since it avoided the particle disorder problem, thereby allowing for a smooth vorticity field which helped with the numerical results. The tensile instability control (TIC) [5] which the scheme employs, prevents the tensile instability problem.

The scheme also employs a particle repositioning step, similar to the Particle Shifting Technique of Lind, S.J., et al. [6], which helps obtain a smoother vorticity field. In order to accelerate the computational time of the simulation, the technique of adaptive particle refinement (APR) is adopted. The lateral wall boundaries were implemented using a *fixed ghost particle* technique, with a free-slip boundary condition.

3 Swimmer Model

3.1 Foil Model

The body of the fish, Saithe (*Pollachius virens*) - Fig 3 is modelled after a NACA0012 foil. The foil is defined as [7], [8]:

$$y_t = \pm 5 T_k c \left[0.2969\sqrt{x'} - 0.1260x' - 0.3516(x')^2 + 0.2843(x')^3 - 0.1015(x')^4 \right] \tag{1}$$

$$x' = \frac{x}{c} \tag{2}$$

where,

- $T_k = 0.12 \rightarrow$ The maximum thickness as a fraction of the chord
- $c = 0.37m \rightarrow$ Chord length
- $x \rightarrow$ Position along the chord
- $y_t \rightarrow$ Half thickness at a given value of x (centerline to surface)

Figure 3: *Pollachius virens*

3.2 Undulation Model

The deformation of the fish is described as the undulation of the centerline, which is approximately fitted as follows [9]:

$$h(x,t) = h_{max}.a_1(x).a_2(t).\sin\left[2\pi\left(\frac{x}{\lambda} - \frac{t}{T}\right)\right] \tag{3}$$

where,

$$a_1 = \begin{cases} 0 & ,x \in [0, 0.2L] \\ \left(\frac{x/L-0.2}{1-0.2}\right)^2 & ,x \in (0.2L, L] \end{cases} \tag{4}$$

$$a_2(t) = \begin{cases} \frac{t}{t_o} - \frac{1}{2\pi}\sin\left(\frac{2\pi t}{t_o}\right) & ,t \in [0, t_o] \\ 1 & ,t \in (t_o, \infty) \end{cases} \tag{5}$$

- $L = 0.37m \rightarrow$ Body length (same as the chord length)
- $T = 0.278s \rightarrow$ Cycle of undulation
- $\lambda = 1.04L \rightarrow$ Wavelength of undulation
- $h_{max} = 0.083L \rightarrow$ Maximum half-amplitude of the deflection at tail tip

This model causes the posterior part of the fish to undulate from rest ($t = 0$) to constant periodic undulation ($t > t_o = 1.0T$).
The phase speed is defined as follows:

$$c_p = \frac{\lambda}{T} \tag{6}$$

The flapping amplitude $A(x,t) = h_{max}a_1(x)a_2(t)$ is determined by the swimming characteristics of a given fish - Fig 4

4 Governing Equations

The governing equations of the $\delta+$ - SPH scheme are as follows:

4.1 Equation of State

$$p_i = c_o^2(\rho_i - \rho_o) \tag{7}$$

where,

1. ρ_o - The reference density when pressure is zero initially
2. c_o - The artificial speed that is based on the weakly-compressible hypothesis [10]. Here, c_o is a constant in the whole simulation and it is determined as: $15U$, where U is the reference velocity [1]

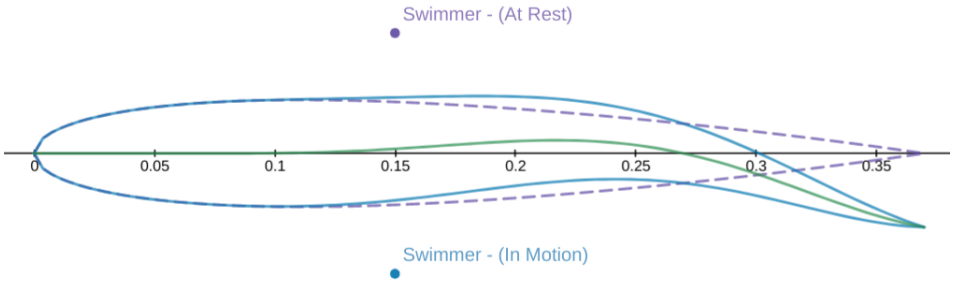


Figure 4: Swimmer

4.2 Renormalization Tensor

$$\mathbb{L}_i = \left[\sum_j \mathbf{r_{ji}} \otimes \nabla_\mathbf{i} \mathbf{W_{ij}} V_j\right]^{-1} \tag{8}$$

$$\lambda_i = \min\left(\text{eigenvalue}(\mathbb{L}_i^{-1})\right) \tag{9}$$

Refer - [10], [11]

4.3 Continuity Equation

$$\frac{D\rho_i}{Dt} = \sum_j \rho_i \mathbf{u_{ij}}.\nabla_\mathbf{i} \mathbf{W_{ij}} V_j + \delta h c_o \Psi_{ij} \frac{\mathbf{r_{ji}}.\nabla_\mathbf{i} \mathbf{W_{ij}}}{|\mathbf{r_{ji}}|^\mathbf{2}} V_j \tag{10}$$

where,

$$\Psi_{ij} = 2(\rho_j - \rho_i) \tag{11}$$

$$V_j = \frac{m_j}{\rho_j} \tag{12}$$

1. δ - The density diffusion parameter $(= 0.1)$ [1]

Refer - [12]

4.4 Continuity Equation - RDGC

$$\frac{D\rho_i}{Dt} = \sum_j \rho_i \mathbf{u_{ij}}.\nabla_\mathbf{i} \mathbf{W_{ij}} V_j + \delta h c_o \Psi_{ij} \frac{\mathbf{r_{ji}}.\nabla_\mathbf{i} \mathbf{W_{ij}}}{|\mathbf{r_{ji}}|^\mathbf{2}} V_j \tag{13}$$

where,

$$\Psi_{ij} = 2(\rho_j - \rho_i) - (\langle\nabla\rho\rangle_\mathbf{i}^\mathbf{L} + \langle\nabla\rho\rangle_\mathbf{j}^\mathbf{L}).\mathbf{r_{ji}} \tag{14}$$

$$\langle \nabla \rho \rangle_{\mathbf{i}}^{\mathbf{L}} = \sum_j (\rho_j - \rho_i) \mathbb{L}_i . \nabla_{\mathbf{i}} \mathbf{W_{ij}} V_j \tag{15}$$

This equation includes the renormalized density gradient correction (RDGC) term as well [10], [13], [14]

4.5 Momentum Equation

$$\frac{D\mathbf{u_i}}{Dt} = \frac{1}{\rho_i} \sum_j \left(F_{ij} \nabla_{\mathbf{i}} \mathbf{W_{ij}} V_j + K \mu \pi_{ij} \nabla_{\mathbf{i}} \mathbf{W_{ij}} V_j \right) + \mathbf{f_i} \tag{16}$$

where,

$$F_{ij} = \begin{cases} -(p_j + p_i), & p_i \geq 0 \\ -(p_j - p_i), & p_i < 0 \end{cases} \tag{17}$$

$$K = 2(\dim + 2) \tag{18}$$

$$\pi_{ij} = \frac{\mathbf{u_{ji}} . \mathbf{r_{ji}}}{|\mathbf{r_{ji}}|^2} \tag{19}$$

1. $\mathbf{f_i}$ - Body-forces
2. μ - The dynamic viscosity ($\mu = \rho_o \nu$), where ν is the kinematic viscosity [1], [14]

4.6 Displacement Equation

$$\frac{D\mathbf{r_i}}{Dt} = \mathbf{u_i} \tag{20}$$

5 Particle Shifting Technique

Once the particle positions are advected through the time, a repositioning is performed as follows [15]:

$$\mathbf{r_i}^* = \mathbf{r_i} + \delta \hat{\mathbf{r}}_\mathbf{i} \tag{21}$$

$$\delta \hat{\mathbf{r}}_\mathbf{i} = \begin{cases} 0 & , \lambda_i \in [0, 0.4) \\ (\mathbb{I} - \mathbf{n_i} \otimes \mathbf{n_i}) \delta \mathbf{r_i} & , \lambda_i \in [0.4, 0.75] \\ \delta \mathbf{r_i} & , \lambda_i \in (0.75, 1] \end{cases} \tag{22}$$

$$\delta \mathbf{r_i} = -CFL.Ma.(2h_{ij})^2 . \sum_j \left[1 + R \left(\frac{W_{ij}}{W(\Delta p)} \right)^n \right] \nabla_{\mathbf{i}} \mathbf{W_{ij}} \varphi_{ij} \left(\frac{m_j}{\rho_i + \rho_j} \right) \tag{23}$$

$$\mathbf{n_i} = \frac{\langle \nabla \lambda_\mathbf{i} \rangle}{|\langle \nabla \lambda_\mathbf{i} \rangle|} \tag{24}$$

$$\langle \nabla \lambda_{\mathbf{i}} \rangle = -\sum_j (\lambda_j - \lambda_i) \mathbb{L}_i \nabla_{\mathbf{i}} \mathbf{W_{ij}} V_j \tag{25}$$

1. Δp = Average particle spacing in the neighbourhood of i
2. $R = 0.2, n = 4$, the same values as suggested by Monaghan [16] are used in $\delta+$ SPH scheme as well
3. $\varphi = 1, h_{ij} = h$, for uniform particle resolution
4. $\frac{\delta \mathbf{r_i}}{\Delta x_i} < 0.05$, using a Wendland C2 kernel with $\frac{h}{\Delta x_i} = 2$
5. $\frac{\delta \mathbf{r_i}}{\Delta x_i} \to 0$ when $\Delta x_i \to 0$

Though this displacement is very small, the gain on the overall accuracy of the simulation is substantial. Further, since the velocity and density fields on the particles are not changed by the PST, the conservation properties of the particle system are in principle preserved.

6 Boundary Conditions

Solid wall boundaries are modelled using the **'Fixed Ghost Particle Technique'** [5]. The pressure of the ghost particle is interpolated from the fluid using the Shepard kernel as follows:

$$p_j = \frac{\sum_i (p_i + \rho_i (\mathbf{a_j} - \mathbf{f}).\mathbf{r_{ij}}) W_{ij}}{\sum_i W_{ij}} \tag{26}$$

where,

1. $j \in$ ghost
2. $i \in$ fluid
3. $\mathbf{a_j}$ = acceleration of ghost particle
4. $\mathbf{f}$ = body-force

In a *"free-slip"* boundary condition the viscous force between the fluid particle and the ghost particle gets forced to zero. The interpolation of the velocity to the ghost particles is not needed since the own velocity u_j of the ghost particle is used. With this solid wall boundary treatment, the evaluation of the normals on the solid surface can be avoided and therefore it is more convenient to model 3D boundaries with complex configurations.

7 Forces & Torques

7.1 Force

The net global force exerted by fluids on solids can be calculated as follows [14]:

$$\mathbf{F}_{f-s} = \sum_i \Big[\sum_j [-(p_i + p_j) + \mu\pi_{ij}] \nabla_\mathbf{i} \mathbf{W_{ij}} V_i V_j \Big] \tag{27}$$

where,

1. $i \in$ fluid
2. $j \in$ solid
3. $(p_i + p_j) \rightarrow$ Pressure component
4. $\mu\pi_{ij} \rightarrow$ Viscous component of the stress tensor

This implementation does not require interpolation on body nodes.

7.2 Torque

The net global torque exerted by fluids on solids can be calculated as follows [14]:

$$\mathbf{T}_{f-s} = \sum_i \left[\sum_j \left((\mathbf{r_j} - \mathbf{r_c}) \times \left[\left(-p_i + \frac{\mu\pi_{ij}}{2} \right) \nabla_\mathbf{i} \mathbf{W_{ij}} \right] + (\mathbf{r_i} - \mathbf{r_c}) \times \left[\left(-p_j + \frac{\mu\pi_{ij}}{2} \right) \nabla_\mathbf{i} \mathbf{W_{ij}} \right] \right) V_i V_j \right] \tag{28}$$

where,

1. $\mathbf{r_c} \rightarrow$ A fixed point *(eg: center of mass)*

7.3 Fluid-Body Coupling

The governing equations of motion are as follows [14]:

$$M \frac{d\vec{\mathbf{V}}_\mathbf{c}}{dt} = M\mathbf{f} + \mathbf{F}_{f-s} \tag{29}$$

where,

1. $\mathbf{V_c} \rightarrow$ Velocity of the center of mass
2. $M \rightarrow$ Mass of the body

3. $\mathbf{f} \rightarrow$ Body-force

$$\frac{d}{dt}\Big(\mathbb{I}_c(t)\omega_{\mathbf{c}}(\mathbf{t})\Big) = \mathbf{T}_{f-s}.\hat{\mathbf{k}} \tag{30}$$

where,

1. $\mathbb{I}_c(t) \rightarrow$ Moment of inertia wrt. center of Mass
2. $\omega_{\mathbf{c}}(\mathbf{t}) \rightarrow$ Angular velocity wrt. center of Mass

The alternative force and torque modelling is given by Monaghan [2] will need to be compared to the above formulation and analysed further.

8 Taylor-Green Vortex

The 2-dimensional decaying vortex (Fig 5) is defined in a square domain of [17]:

$$x, y \in [0, \pi]$$

The governing equations are as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{31}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{32}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{33}$$

The Taylor-Green vortex solution is given by:

$$\begin{aligned} u &= \sin(x)\cos(y).F(t) \\ v &= -\cos(x)\sin(y).F(t) \\ F(t) &= e^{-2\nu t} \\ p &= \frac{\rho}{4}\big(\cos(2x) + \sin(2y)\big).F^2(t) \end{aligned}$$

A periodic boundary condition is used, to solve the problem.

8.1 PySPH Implementation

The Taylor-Green vortex problem was implemented using the PySPH framework (Fig 6) [18].

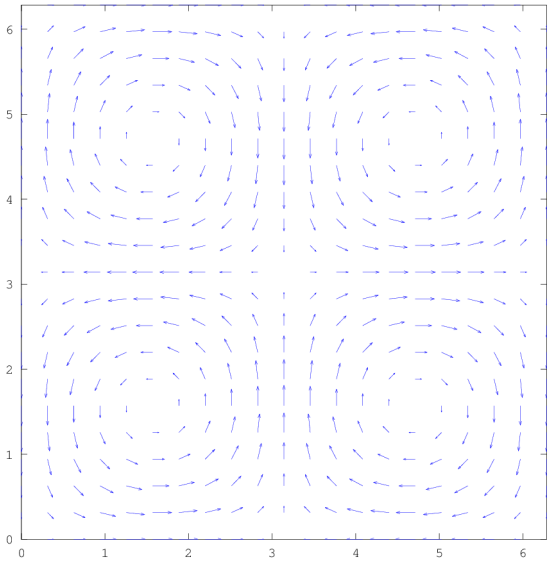
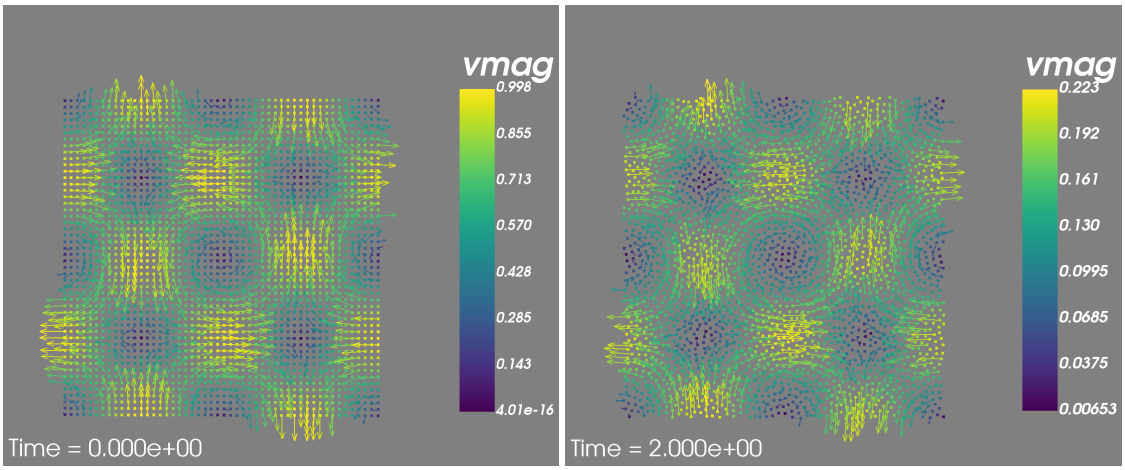


Figure 5: TGV: Vector plot



(a) $t = 0$ (b) $t = 0$

Figure 6: TGV: Vector plot, $\delta+$ - SPH

8.1.1 Iterations of the $\delta+$ - SPH scheme

The problem was implemented in the following way:

1. Iteration 1:
   (a) Kernel - Quintic Spline
   (b) Euler Integrator

(c) Periodic Boundary Conditions

(d) Equation of State

(e) Continuity Equation (**without RDGC**)

(f) Momentum Equation

2. Iteration 2:

(a) Kernel - Quintic Spline

(b) Euler Integrator

(c) Periodic Boundary Conditions

(d) Equation of State

(e) Renormalization Tensor

(f) Continuity Equation (**with RDGC**)

(g) Momentum Equation

3. Iteration 3:

(a) Kernel - Wendland Quintic (C2)

(b) Euler Integrator

(c) Periodic Boundary Conditions

(d) Equation of State

(e) Renormalization Tensor

(f) Continuity Equation (**with RDGC**)

(g) Momentum Equation

(h) Particle Shifting Technique

The error quantities are described as below:

1. **Decay**:

$$D(t) = max(|\mathbf{u}(\mathbf{t})|) \tag{34}$$

$$D^*(t) = max(|\mathbf{u}^*(\mathbf{t})|) \tag{35}$$

2. L_1 **error**:

$$L_1(t) = \frac{avg(|\mathbf{u}(\mathbf{t}) - \mathbf{u}^*(\mathbf{t})|)}{avg(|\mathbf{u}^*(\mathbf{t})|)} \tag{36}$$

3. L_∞ **error**:

$$L_\infty(t) = \left| \frac{max(|\mathbf{u}(t)|) - max(|\mathbf{u}^*(t)|)}{max(|\mathbf{u}^*(t)|)} \right| \tag{37}$$

4. L_1 **error for** p:

$$L_1(t) = \frac{avg(|p(t) - p^*(t)|)}{max(|p^*(t)|)} \tag{38}$$

where, all $(.)^*$ quantities represent the expected values, from the analytical solution, and non-$(.)^*$ quantities represent the actual values, obtained from numerical solution, and:

$$avg(\mathbf{x}) = \frac{\sum_i^N \mathbf{x_i}}{N} \tag{39}$$

$$max(x) = max\Big(x_1, x_2, x_3, x_4, ..., x_N\Big) \tag{40}$$

Result

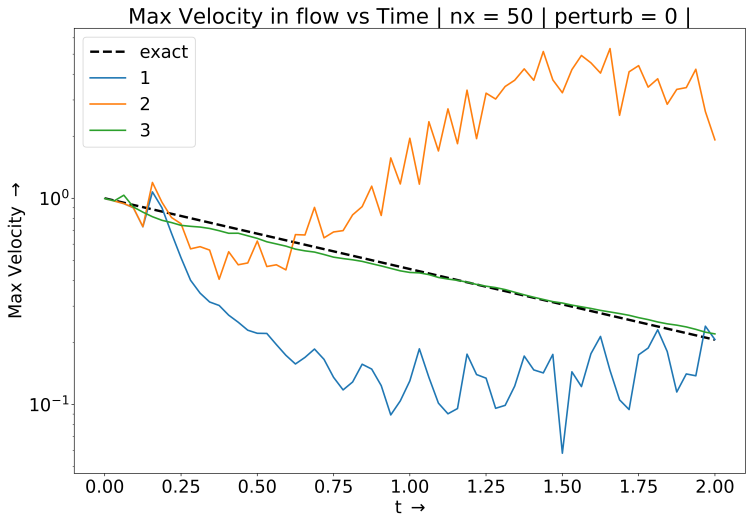


Figure 7: TG: Iterations - Decay

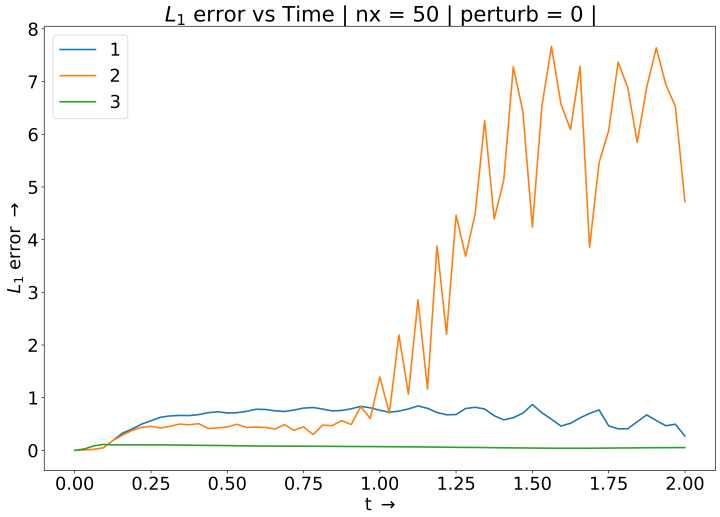


Figure 8: TG: Iterations - L_1 Error

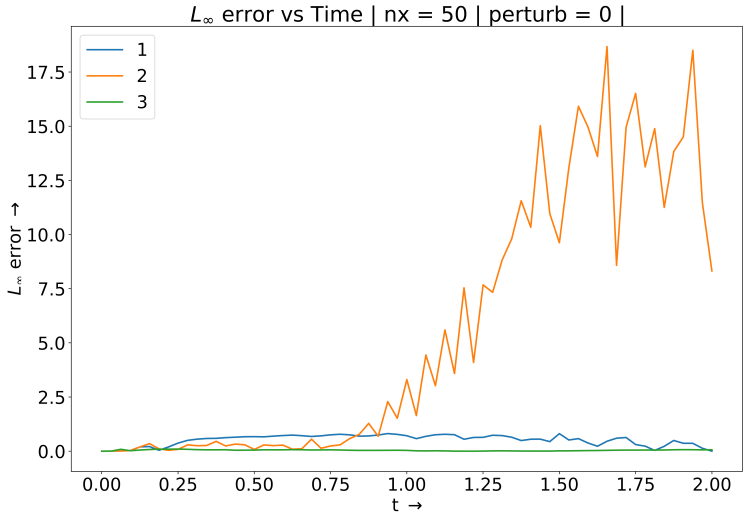


Figure 9: TG: Iterations - L_∞ Error

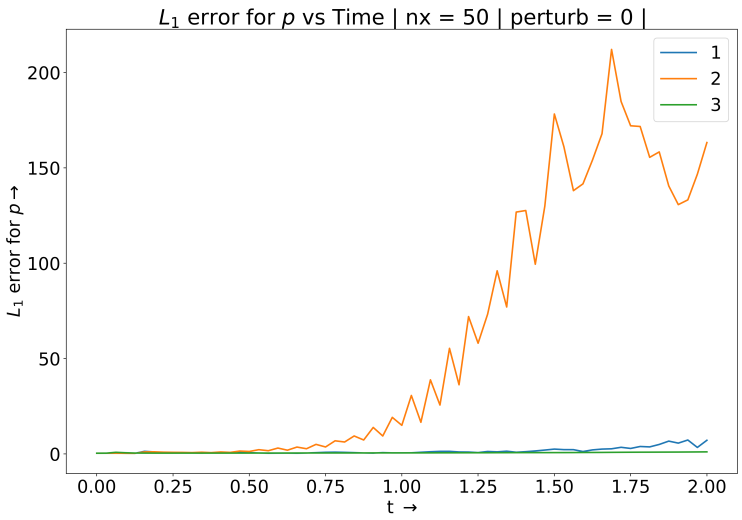


Figure 10: TG: Iterations - L_1 Error for p

Based on the error plots, the following was observed:

1. The error was the least in Iteration-3, which had all the elements of $\delta+$ - SPH scheme for a boundary-less problem
2. While using the Particle Shifting Technique, the $\delta\mathbf{r}$, which was being calculated was diverging ($|\delta\mathbf{r}|/\Delta x_i > 10^4$), which was unacceptable for the simulation. Hence the following modifications were performed to the PST function:
   (a) Replace the Quintic Spline kernel with the Wendland Quintic (C2) kernel
   (b) Modify the CFL constant appropriately
   (c) Manually set the positional shift of all those particles which have $\delta\mathbf{r}/\Delta x_i > 0.075$ to zero. The rationale behind this step, can be explained from the fact that heuristically , in all the simulations performed by Sun, P.N, et al. [15], the ratio $\delta\mathbf{r}/\Delta x_i$ was less than 0.05

8.1.2 Comparison with other schemes

The $\delta+$ - SPH scheme that is implemented is compared with some of the already available SPH schemes on PySPH - EDAC, TVF, and WCSPH. The results are as follows:

Low resolution, Equispaced-particles

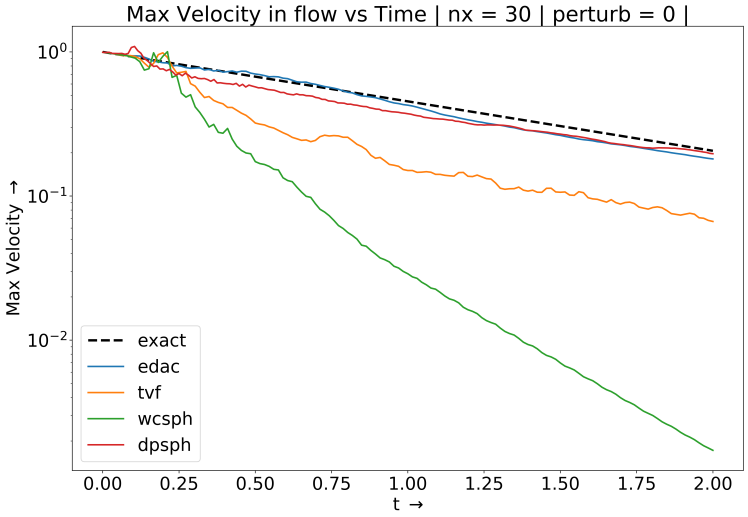


Figure 11: TG:Decay | nx = 30 | perturb = 0|

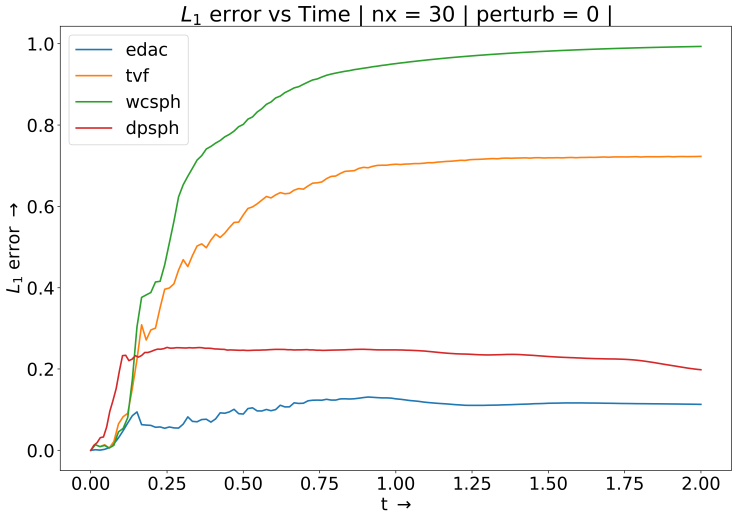


Figure 12: TG: L_1 Error | nx = 30 | perturb = 0|

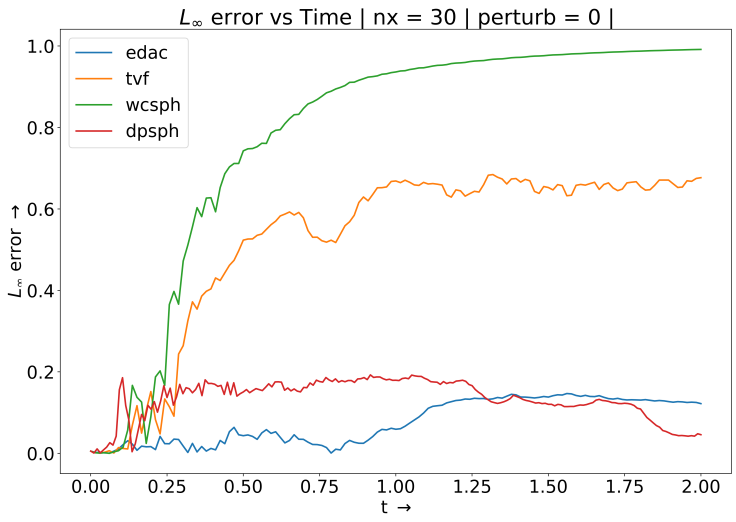


Figure 13: TG: L_∞ Error | nx = 30 | perturb = 0|

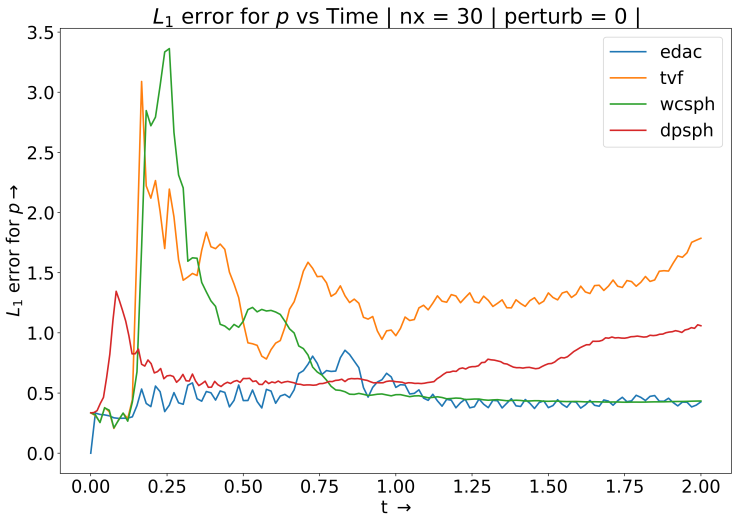


Figure 14: TG: L_1 Error for p | nx = 30 | perturb = 0|

Low resolution, Perturbed-particles

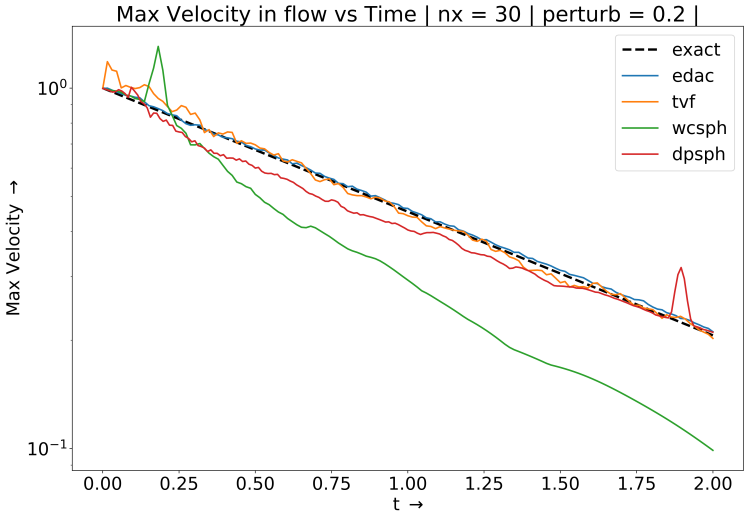


Figure 15: TG: Decay | nx = 30 | perturb = 0.2|

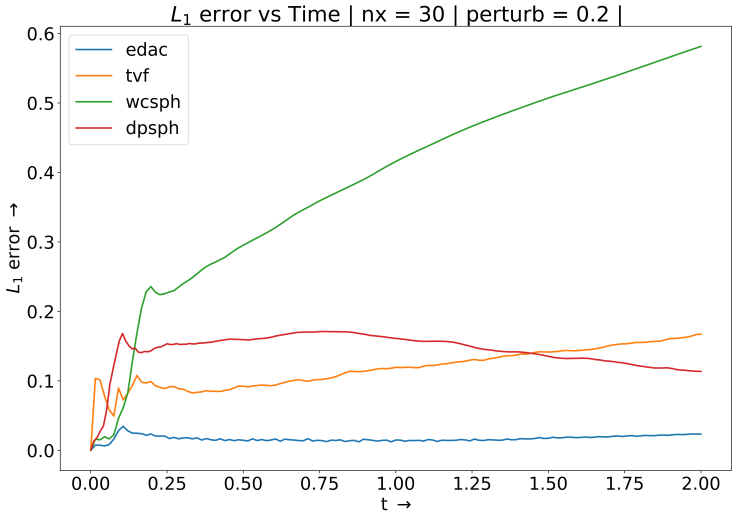


Figure 16: TG: L_1 Error | nx = 30 | perturb = 0.2|

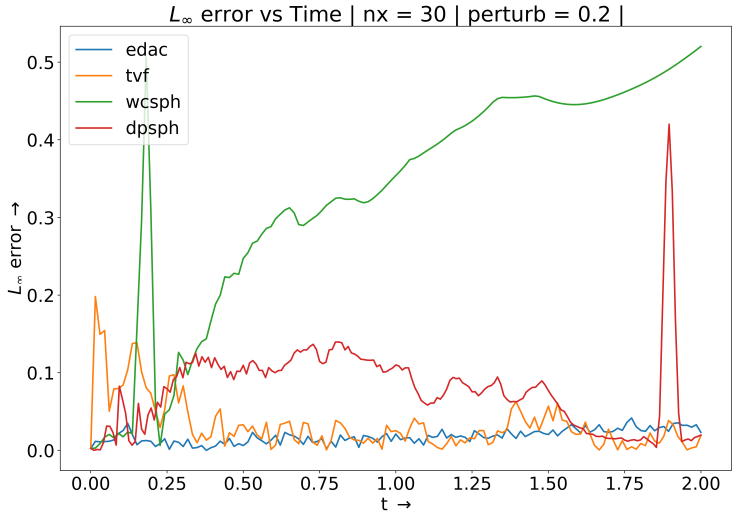


Figure 17: TG: L_∞ Error | nx = 30 | perturb = 0.2|

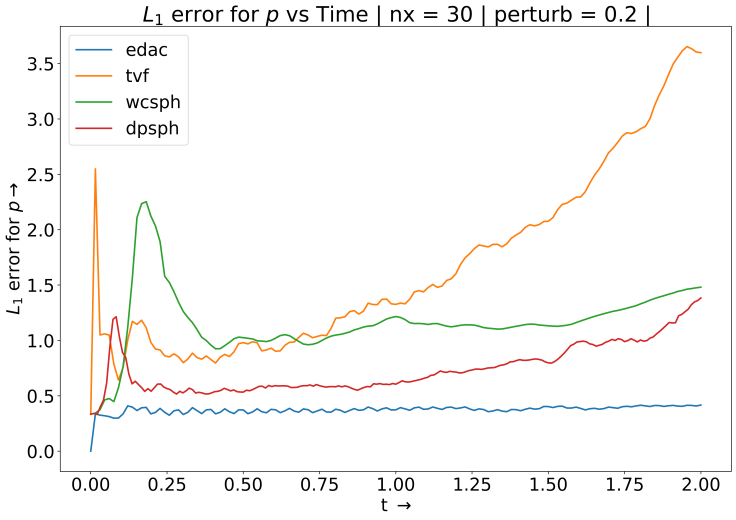


Figure 18: TG: L_1 Error for p | nx = 30 | perturb = 0.2|

High resolution, Equispaced-particles

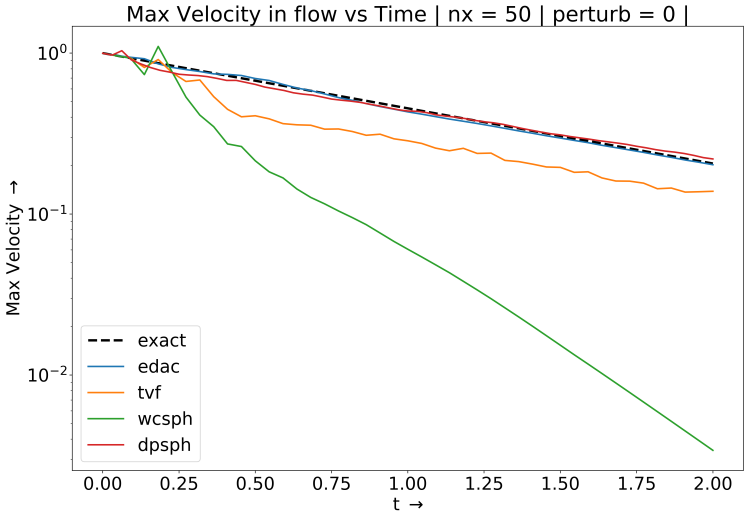


Figure 19: TG: Decay | nx = 50 | perturb = 0|

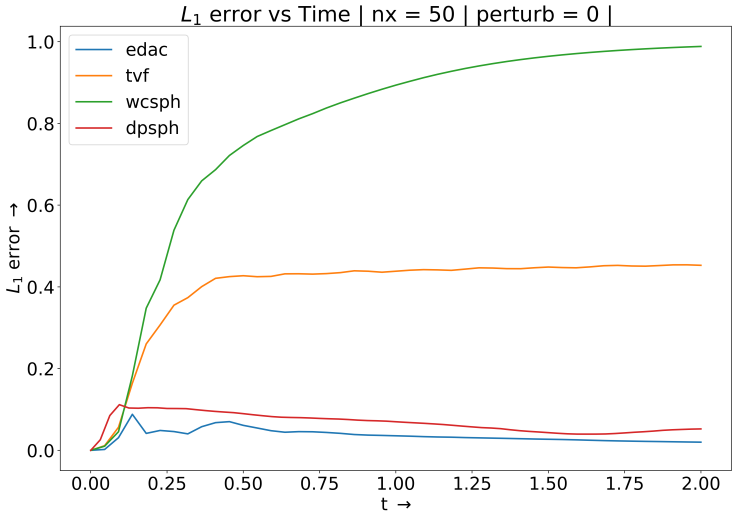


Figure 20: TG: L_1 Error | nx = 50 | perturb = 0|

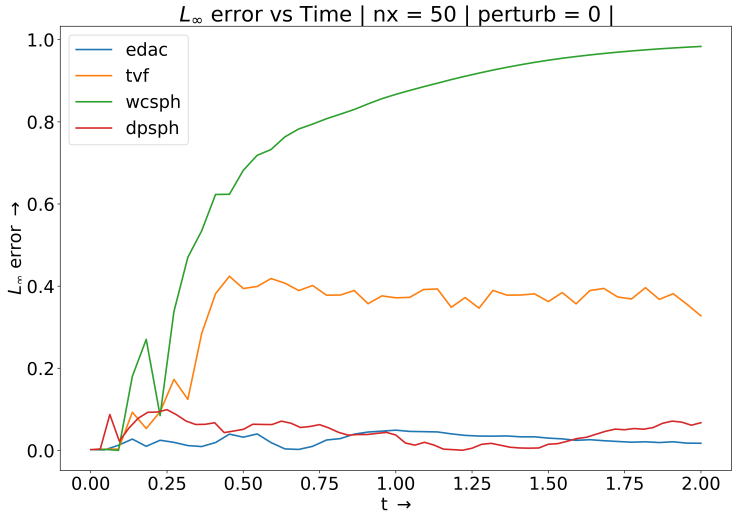


Figure 21: TG: L_∞ Error | nx = 50 | perturb = 0|

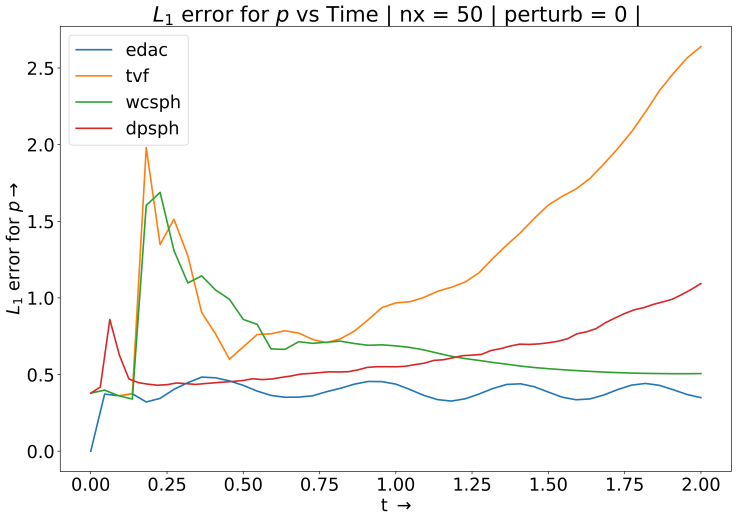


Figure 22: TG: L_1 Error for p | nx = 50 | perturb = 0|

High resolution, Perturbed-particles

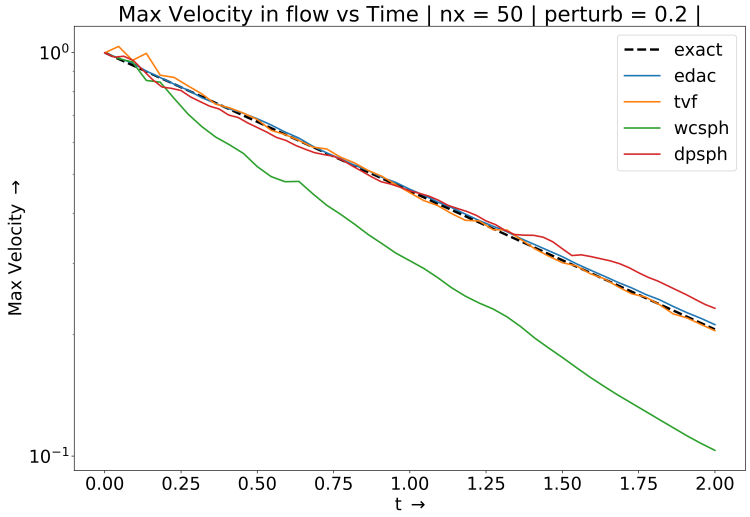


Figure 23: TG: Decay | nx = 50 | perturb = 0.2|

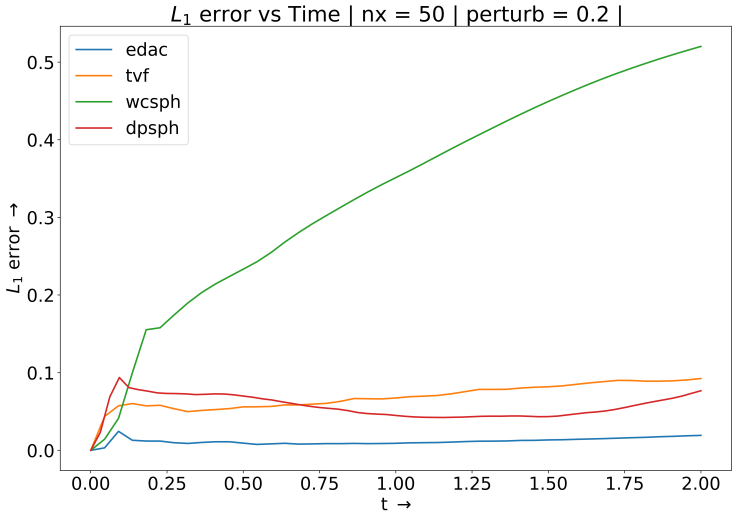


Figure 24: TG: L_1 Error | nx = 50 | perturb = 0.2|

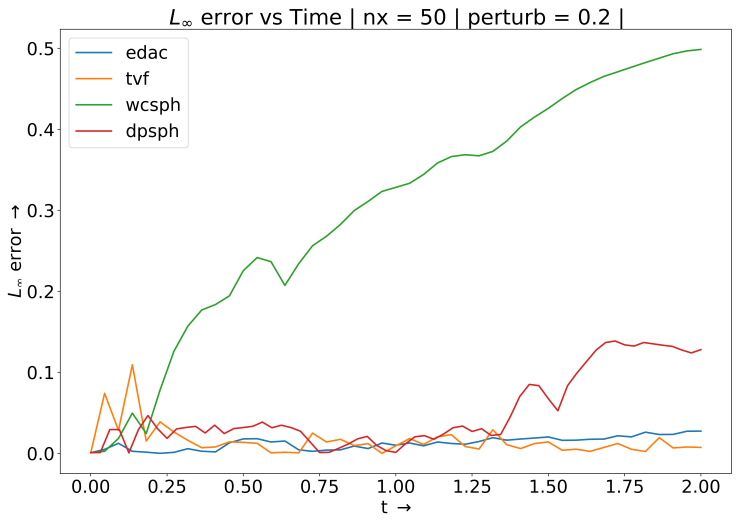


Figure 25: TG: L_∞ Error | nx = 50 | perturb = 0.2|

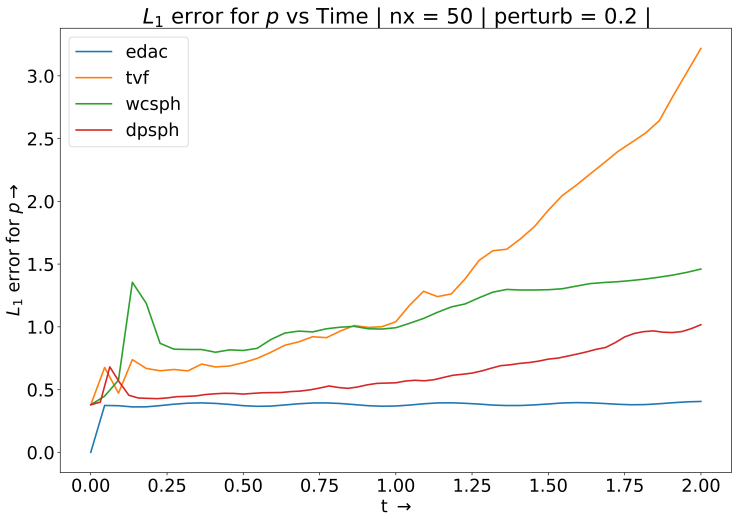


Figure 26: TG: L_1 Error for p | nx = 50 | perturb = 0.2|

The following observations were made, based on the error plots that were generated:

1. The EDAC scheme had the least error, amongst all the schemes which were taken into consideration

2. $\delta+$ - SPH scheme was at par, or just below par with EDAC in terms of accuracy in most of the error types

3. Each simulation was run three times, and the average run-time was noted, based on which, one can observe that for a resolution of $nx = 30$ (*low resolution*), $\delta+$ - SPH scheme had the longer runtime, while for the case of $nx = 50$(*high resolution*), EDAC had the higher runtime - Fig 27

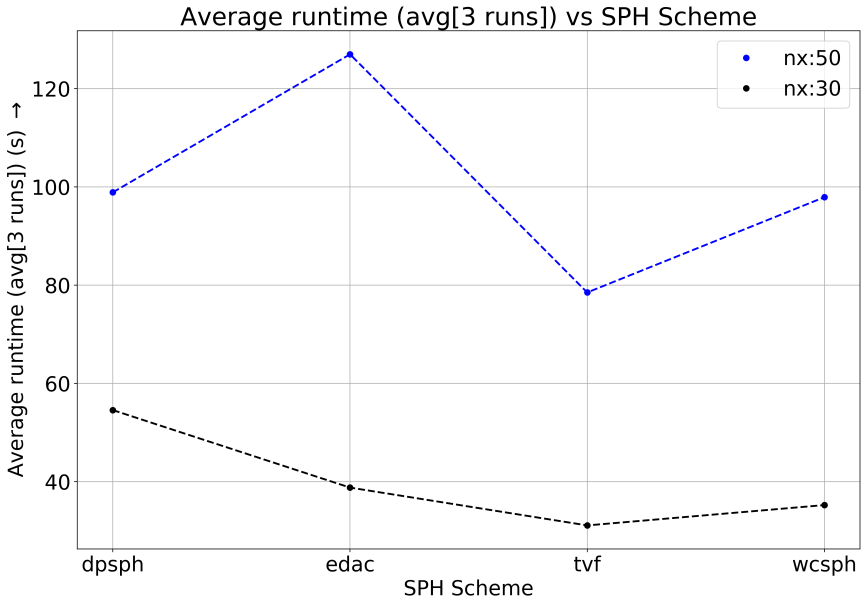


Figure 27: SPH Schemes - Runtime

9 Conclusions

The $\delta+$ - SPH scheme, having only been implemented on a simple, boundary-less problem showed promising results. However, it will have to be studied with complex benchmark problems, which involve boundaries, and a higher *Re* flow, before further, generalized comments can be made on the scheme. Also with further improvements, the runtime can be shortened (*Eg: better implementation of the Renormalization Tensor*), and the accuracy of the scheme can be improved with a more robust PST implementation. The adaptive particle refinement (APR) was not explored in-depth over the course of this project, as it is a non-trivial

exercise. However, since it is a crucial part of the $\delta+$ - SPH scheme, it cannot be neglected in future studies.

The force and torque formulations were not studied with respect to the Monaghan formulation [2], and thereby the selection of an appropriate formulation for the given problem statement, was left incomplete. Future work in this task will require the implementation of the *fixed ghost particle* technique for the boundary conditions, which the Bouscasse, B, et al. [14] formulation is based on.

Future Work

Once the $\delta+$ - SPH scheme is complete, the following domains can be explored to solve currently unanswered questions:

1. Optimize the SPH scheme. [14]
2. Include Torque effects, as well as the transversal and rotational motions of the foil which are ignored. [1]
3. Extend $\delta+$ - SPH scheme to fish-like swimming problems in 3D.
4. Explain the mechanism of fish schooling in nature, by investigating the complex interactions among more than two fishes in different positions.
5. Study the interaction between the swimmer and free-surfaces.
6. Compare the results of δ - SPH and $\delta+$ - SPH scheme; verify the superior accuracy and stability of the $\delta+$ - SPH, since introduction of PST causes the loss of exact momenta and energy conservation. [15]
7. Water entry of spheres or cylinders at higher entry speeds can be conducted. For these cases, the air phase has to be considered and a multi-phase SPH model capable of considering the real sound speed of the air and the water will be needed.
   If the water entry speed is large enough, then cavitation will be generated. This can be solved with a cavitation model [5]
8. Analyze the impact of viscosity and vorticity generation [13]

References

[1] P.-N. Sun, A. Colagrossi, and A.-M. Zhang, "Numerical simulation of the self-propulsive motion of a fishlike swimming foil using the δ -sph model", *Theoretical and Applied Mechanics Letters*, vol. 8, no. 2, pp. 115–125, 2018. DOI: 10.1016/j.taml.2018.02.007.

[2] J. J. Monaghan, "Smoothed particle hydrodynamics and its diverse applications", *Annual Review of Fluid Mechanics*, vol. 44, pp. 323–346, 2012.

[3] J. Kajtar and J. J. Monaghan, "Sph simulations of swimming linked bodies", *Journal of Computational Physics*, vol. 227, no. 19, pp. 8568–8587, 2008.

[4] P. Sun, A. Colagrossi, S. Marrone, and A. Zhang, "Detection of lagrangian coherent structures in the sph framework", *Computer Methods in Applied Mechanics and Engineering*, vol. 305, pp. 849–868, 2016.

[5] P. Sun, A.-M. Zhang, S. Marrone, and F. Ming, "An accurate and efficient sph modeling of the water entry of circular cylinders", *Applied Ocean Research*, vol. 72, pp. 60–75, 2018. DOI: 10.1016/j.apor.2018.01.004.

[6] S. J. Lind, R. Xu, P. K. Stansby, and B. D. Rogers, "Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves", *Journal of Computational Physics*, vol. 231, no. 4, pp. 1499–1523, 2012.

[7] *Naca airfoil*, Jun. 2020. [Online]. Available: https://en.wikipedia.org/wiki/NACA_airfoil#Equation_for_a_symmetrical_4-digit_NACA_airfoil.

[8] L. Charles L., J. Cuyler W. Brooks, H. Acquilla S., and S. Darrell W., "Computer program to obtain ordinates for naca airfoils", Tech. Rep., 1996.

[9] Y. Yan, W. Guan-Hao, Y. Yong-Liang, and T. Bing-Gang, "Two-dimensional self-propelled fish motion in medium: An integrated method for deforming body dynamics and unsteady fluid dynamics", *Chinese Physics Letters*, vol. 25, no. 2, pp. 597–600, 2008. DOI: 10.1088/0256-307x/25/2/066.

[10] P. Sun, A. Colagrossi, S. Marrone, M. Antuono, and A.-M. Zhang, "A consistent approach to particle shifting in the δ-plus-sph model", *Computer Methods in Applied Mechanics and Engineering*, vol. 348, pp. 912–934, 2019.

[11] S. Marrone, A. Colagrossi, D. L. Touzé, and G. Graziani, "Fast free-surface detection and level-set function definition in sph solvers", *Journal of Computational Physics*, vol. 229, no. 10, pp. 3652–3663, 2010. DOI: 10.1016/j.jcp.2010.01.019.

[12] M. Antuono, A. Colagrossi, S. Marrone, and D. Molteni, "Free-surface flows solved by means of sph schemes with numerical diffusive terms", *Computer Physics Communications*, vol. 181, no. 3, pp. 532–549, 2010. DOI: 10.1016/j.cpc.2009.11.002.

[13] S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. L. Touzé, and G. Graziani, "δ-sph model for simulating violent impact flows", *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 13-16, pp. 1526–1542, 2011. DOI: 10.1016/j.cma.2010.12.016.

[14] B. Bouscasse, A. Colagrossi, S. Marrone, and M. Antuono, "Nonlinear water wave interaction with floating bodies in sph", *Journal of Fluids and Structures*, vol. 42, pp. 112–129, 2013. DOI: 10.1016/j.jfluidstructs.2013.05.010.

[15] P. Sun, A. Colagrossi, S. Marrone, and A. Zhang, "The δ+-sph model: Simple procedures for a further improvement of the sph scheme", *Computer Methods in Applied Mechanics and Engineering*, vol. 315, pp. 25–49, 2017. DOI: 10.1016/j.cma.2016.10.028.

[16] J. J. Monaghan, "Sph without a tensile instability", *Journal of computational physics*, vol. 159, no. 2, pp. 290–311, 2000.

[17] A. Salih, *Taylor–green vortex*, Indian Institute of Space Science and Technology, Thiruvananthapuram, Feb. 2011.

[18] P. Ramachandran, K. Puri, A. Bhosale, A. Dinesh, A. Muta, P. Negi, R. Govind, S. Sanka, P. Pandey, C. Kaushik, *et al.*, "Pysph: A python-based framework for smoothed particle hydrodynamics", *arXiv preprint arXiv:1909.04504*, 2019.