



**FACULTY  
OF SCIENCE**

# **Differential Equations in Astrophysics**

**Andreas Pettersson**

**in Collaboration with Leo Bublies**

# List of Abbreviations

**SPH** Smoothed Particle Hydronamics

# Contents

<b>List of Abbreviations</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Theory</b>	<b>1</b>
2.1 Smoothing kernel . . . . .	1
2.2 Sod’s shock tube . . . . .	2
2.3 Planet collision . . . . .	3
<b>3 Method</b>	<b>3</b>
3.1 Sod’s Shock Tube . . . . .	3
<b>4 Results</b>	<b>4</b>
4.1 Sod’s Shock Tube . . . . .	4
4.1.1 Planetary Collisions . . . . .	4
<b>5 Conclusion</b>	<b>4</b>
<b>References</b>	<b>6</b>

# 1 Introduction

**Smoothed Particle Hydronamics (SPH)** is an efficient way for simulating continuous systems, such as fluid dynamics. It is constructed in a meshfree grid of particles, where the interactions between nearest neighbours are determined by a smoothing kernel. The smoothing kernel itself is generally given as a cubic spline or occasionally as a Gaussian distribution; as the form of the kernel is dependent on the problem at hand, there is no universally accepted approach.

In this report the **SPH** method will be used to simulate Sod's shock tube and two gas-like planets colliding.

Sod's shock tube is a widely used computational problem that is used to assert correctly written — as well as the efficiency — of written **SPH** code. Since the same set of main equations are used to simulate the two gas-like planets, Sod' shock tube problem is appropriately chosen to ascertain working code, as mentioned.

## 2 Theory

Here the theory — and mainly equations — are given for each project; being Sod' shock tube and the collision of two gas-like planets.

### 2.1 Smoothing kernel

The smoothing kernel, or simply kernel, is denoted  $W(R, h)$ . In the two simulations mentioned, the following form is used:

$$W = \alpha_d \times \begin{cases} \frac{2}{3} - R^2 + \frac{1}{2}R^3, & 0 \leq R < 1 \\ \frac{1}{6}(2 - R)^3, & 1 \leq R < 2, \\ 0, & R \geq 2 \end{cases} \quad (1)$$

where  $R_{ij} = r_{ij}/h = |\mathbf{x}_i - \mathbf{x}_j|/h = |d\mathbf{x}|/h$ , with  $h$  being the smoothing length. In addition, the gradient of the kernel — which is naturally used in the equations — is given by

$$\nabla W(R, h) = \begin{cases} \alpha_d \times \left(-2 + \frac{3}{2}R\right) \frac{d\mathbf{x}}{h^2}, & 0 \leq R < 1 \\ -\alpha_d \times \frac{1}{2}(2 - R)^2 \frac{d\mathbf{x}}{h^2}, & 1 \leq R < 2. \\ 0, & R \geq 2 \end{cases} \quad (2)$$

The quantity  $\alpha_d$  is dependent on the dimension  $d$  and is given by

$$\alpha_1 = \frac{1}{h} \quad (3)$$

$$\alpha_2 = \frac{15}{7\pi h^2} \quad (4)$$

$$\alpha_3 = \frac{3}{2\pi h^3}. \quad (5)$$

## 2.2 Sod's Shock Tube

Sod's shock tube is composed of a 1-dimensional 'tube' with particles spread out along the  $x$ -axis. A membrane is placed in the middle of the tube, where each side of the membrane contain a difference in particles and hence density and pressure. Using a summation density approach, the equations used stems from the Navier-Stokes equations, where the density for the  $i$ :th particle is given by

$$\rho_i = \sum_j m_j W_{ij}. \quad (6)$$

Here  $m_j$  is the mass of the  $j$ :th particle and  $W_{ij}$  the smoothing kernel. Furthermore, the pressure is calculated by the equations of state;

$$p_i = (\gamma - 1)\rho_i e_i, \quad (7)$$

where  $e_i$  is the energy and  $\gamma = C_p/C_v = 1.4$  is the ratio between the heat capacity at constant pressure and volume. The rate of change of position is given by

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i, \quad (8)$$

$\mathbf{v}_i$  being the velocity. In turn, the rate of change of velocity is

$$\frac{d\mathbf{v}_i}{dt} = - \sum_{j=1}^N m_j \left( \frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) \nabla_i W_{ij}. \quad (9)$$

Here, the term  $\Pi_{ij}$  is the artificial viscosity; if not added, un-physical results will occur. It is given by

$$\Pi_{ij} = \begin{cases} \frac{-\alpha_{\Pi} \bar{c}_{ij} \phi_{ij} + \beta_{\Pi} \phi_{ij}^2}{\bar{\rho}_{ij}}, & \mathbf{v}_{ij} \cdot \mathbf{x}_{ij} < 0 \\ 0, & \mathbf{v}_{ij} \cdot \mathbf{x}_{ij} \geq 0 \end{cases}. \quad (10)$$

The terms  $\alpha_{\Pi}$ ,  $\beta_{\Pi}$  are constants, while

$$\phi_{ij} = \frac{h_{ij} \mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{|\mathbf{x}_{ij}|^2 + \varphi^2}, \quad (11)$$

where  $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$  and similarly for  $\mathbf{v}_{ij}$ . Furthermore,  $\varphi = 0.1h_{ij}$  and

$$\bar{\rho}_{ij} = \frac{1}{2}(\rho_i + \rho_j), \quad (12)$$

is the average density. Similarly,  $\bar{c}_{ij}$  is the average sound speed, where the sound speed is given by

$$c_i = \sqrt{(\gamma - 1)e_i}, \quad (13)$$

and  $h_{ij}$  is the average smoothing length used in the kernel. Finally, the rate of change of energy is given by

$$\frac{de_i}{dt} = \frac{1}{2} \sum_{j=1}^N m_j \left( \frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) \mathbf{v}_{ij} \cdot \nabla_i W_{ij}. \quad (14)$$

## 2.3 Planet Collision

For the two gas-like planets colliding, the same equations are implemented with the exception of gravity. Thus, in addition to the above stated equations, a gravitational acceleration is added, given by

$$\left( \frac{d\mathbf{v}_i}{dt} \right)_{\text{Gravity}} = -\frac{G}{2} \sum_{j=1}^N m_j (\nabla_i \phi_{ij}(h_i) + \nabla_i \phi_{ij}(h_j)) \frac{\mathbf{x}_{ij}}{r_{ij}}, \quad (15)$$

where  $G$  is the gravitational constant and  $\nabla\phi$  is the gradient of the gravitational potential, given by

$$\frac{\partial\phi(r, h)}{\partial r} = \begin{cases} \frac{1}{h^2} \left( \frac{4}{3}R - \frac{6}{5}R^3 + \frac{1}{2}R^4 \right), & 0 \leq R < 1 \\ \frac{1}{h^2} \left( \frac{8}{3}R - 3R^2 + \frac{6}{5}R^3 - \frac{1}{6}R^4 - \frac{1}{15R^2} \right), & 1 \leq R < 2 \\ \frac{1}{r^2}, & R \geq 2 \end{cases} \quad (16)$$

## 3 Method

Both sub-projects were programmed in python using the numpy and scipy libraries. No for-loops were used, in favor of broadcasting numpy matrices.

### 3.1 Sod's Shock Tube

For Sod's shock tube in 1-D, 400 particles were used, with initial conditions given in the lab manual[1]. The smoothing constant  $h$  giving 5 nearest neighbors was found to be 0.01 through inspecting matrices, and was used throughout the simulations. The integrator used was RK45 with the scipy tool `scipy.integrator.solve_ivp`.

### 3.2 Planet Collision

For planet collisions, 600 particles were used, with 300 per planet. The smoothing constant  $h$  was found to be  $1E7$ .

## 4 Results

Here is presented the results from Sod's shock tube and planetary collisions.

### 4.1 Sod's Shock Tube

Running the simulation with  $dt = 0.05$  and 20 steps, produced the results in **Fig 1**.

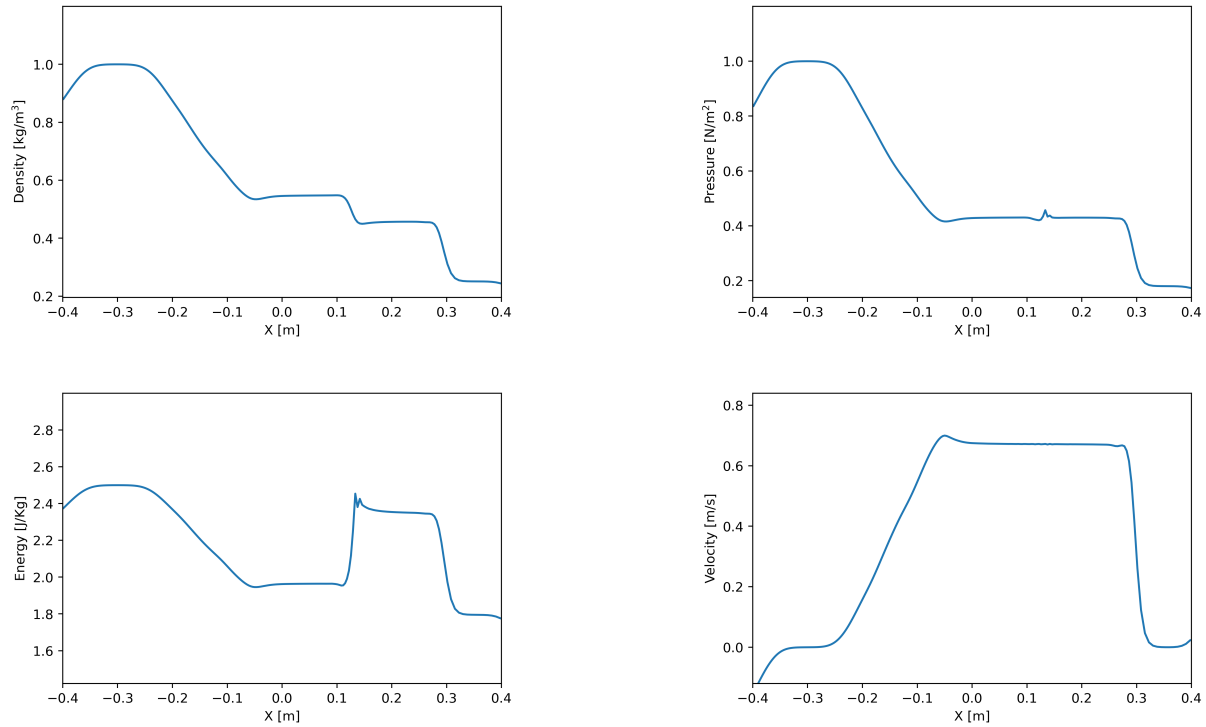


Figure 1: Density, pressure, internal energy and velocity of the system versus the  $x$ -coordinate.

The results are close to the expected values from previous simulations from the lab manual[1].

### 4.2 Planet Collision

Unfortunately, the produced simulation for the planetary collision didn't work as expected.

## 5 Conclusion

The 1-D Sod's shock tube part of the project shows almost exact comparison to the expected values. We conclude that the method implemented does what it should at an acceptable level. Furthermore, the time for simulation is gratifying, w

The planetary collision problem would have required more time for solving bugs in the code. We were able to produce moving particles in a 3-D environment. However it seems that gravitation, although implemented, didn't work as expected, and that the term **REF** was large in comparison. More time would be needed to debug this issue.



## References

- [1] D. Hobbs. *SPH Labs - Lab Manual*. Given for FYSN33 - Computational Physics. Undated.