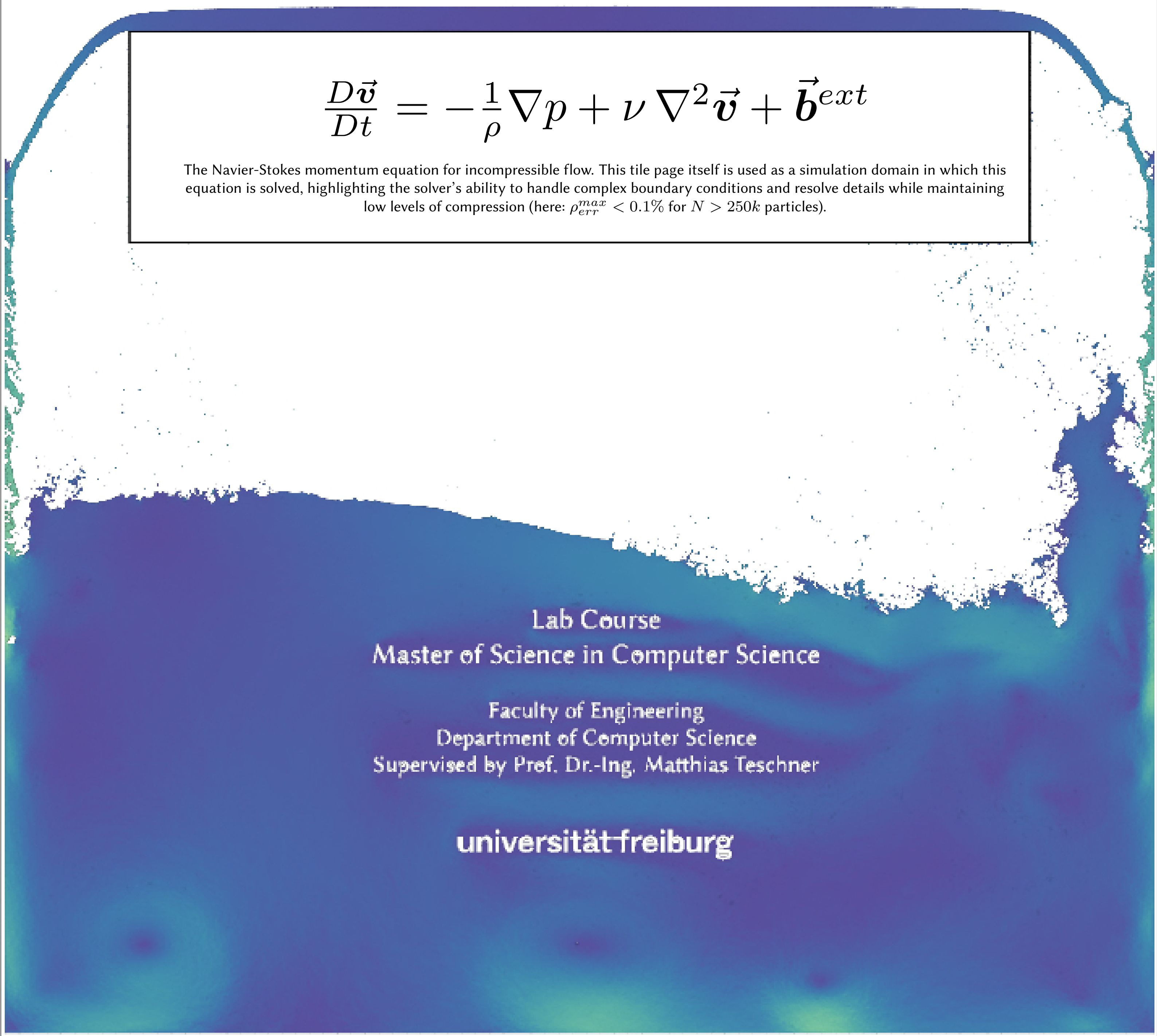
Equation of State Solver for Smoothed Particle Hydrodynamics

Julian Karrer



$$\frac{D\vec{\boldsymbol{v}}}{Dt} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \vec{\boldsymbol{v}} + \vec{\boldsymbol{b}}^{ext}$$

The Navier-Stokes momentum equation for incompressible flow. This tile page itself is used as a simulation domain in which this equation is solved, highlighting the solver's ability to handle complex boundary conditions and resolve details while maintaining low levels of compression (here: $\rho_{err}^{max} < 0.1\%$ for $N > 250k$ particles).

Lab Course
Master of Science in Computer Science

Faculty of Engineering
Department of Computer Science
Supervised by Prof. Dr.-Ing. Matthias Teschner

universität freiburg

Contents

Chapter 1

Introduction

Chapter 2

Governing Equations of Fluid Flow

In an attempt to create a numerical solver for fluid dynamics problems, the governing equations of the underlying physical process must first be understood and formulated. Only then can an appropriate discretization be applied to numerically solve for desired properties of a system. In this chapter, the abstractions of continuum mechanics are used as a framework to describe incompressible flow. Physical principles such as conservation of mass and momentum are used to derive the continuity and momentum equations which encode them, then augmented by constitutive relations which describe properties of Newtonian fluids to finally yield the Navier-Stokes equations as governing equations.[12]

The particular form of these equations will favour a Lagrangian view of the system, in which the frame of reference in which quantities are described is advected along with the flow of the fluid itself, which will seamlessly integrate with the discretization scheme later used to derive workable numerical algorithms.

2.1 Lagrangian and Eulerian Continuum Mechanics

The purpose of our mathematical modelling of fluids is to simulate fluid dynamics at macroscopic scales with numerical methods. We know that fluids consist of innumerable molecules, and smaller yet quarks, interacting in complex ways, which give rise to emergent properties that we observe on a macroscopic scale. Instead of resolving all scales and simulating from quantum mechanical principles up, we content with modelling the emergent properties themselves, focusing on the question of how fluids behave instead of asking why. Our macroscopic scale is so many orders of magnitude larger than the discrete, physical reality, that we can reasonably assume quantities describing the fluid to be continuous and tackle them with the tools of calculus. This gives rise to the field of **Continuum Mechanics**.
In the following derivations, two major points of view can be taken, which produce different but equivalent forms of equations: the Eulerian or conservation forms, and the Lagrangian or nonconservation forms of the equations[1].

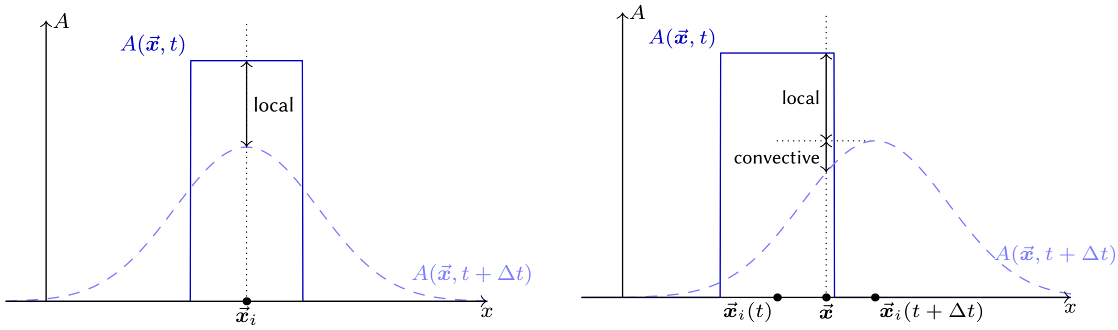
Using the assumption from continuum mechanics that quantities of our fluid are continuously distributed in space and asserting that they be differentiable, we can define derivatives on them. The two major forms of equations arise from a different interpretation of the so-called substantial derivative[1] or material derivative[2] $\frac{D}{Dt}$. This operator describes the instantaneous time rate of change of a quantity of a continuum element as it moves through space[1]. This movement through space however can be observed from different frames of reference:

- a frame that is advected along with the flow of the fluid, in which the continuum element observed is constant
- a frame that is constant in space at a fixed point, observing the flow of the fluid as continuum elements move through it

For both frames of reference, it can be derived that the material derivative in vector notation is[1]:

$$\frac{D}{Dt} = \underbrace{\frac{\partial}{\partial t}}_{\text{local derivative}} + \underbrace{(\vec{v} \cdot \nabla)}_{\text{convective derivative}} \tag{2.1}$$

where $\vec{v}$ is the velocity of the element and ∇ denotes the differential operator $\left(\frac{\partial}{\partial x_0}, \frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n}\right)^T$ in n dimensions[1]. If an Eulerian view is chosen, there is an additional term for the convective derivate, which describes a rate of change of a quantity at a fixed point due to movement of the fluid. If a Lagrangian view is taken, the reference frame is advected with the velocity $\vec{v}$, precisely such that the convective derivative



(a) Lagrangian view (b) Eulerian view

Figure 2.1: A field quantity A in one dimension is shown at some time t and a later time $t + \Delta t$ where the distribution of A has changed due to some diffusive process and the quantity was advected in positive x-direction. In the Lagrangian view, changes in a quantity A are evaluated at the advected position $\vec{x}_i$ at any point in time and only the local derivative is needed to describe the change in A. In the Eulerian view there are two reasons for A at a point $\vec{x}$ fixed in space to change: the local derivative due to the diffusive process and the convective derivative due to the advection of the quantity with the velocity field.

is zero and the material derivative simply becomes the total time derivative of a quantity. The difference between the two views is illustrated in Figure 2.1. In a way, the Lagrangian frame of reference is chosen precisely such that only local derivatives suffice to describe material derivatives by using a coordinate transformation defined by the velocity field.

How this simpler, Lagrangian form can be used largely depends on the later choice of discretization: discretizing space and tracking the fluid that moves through it favours an Eulerian framework, while discretizing the continuum into particles and sampling quantities only at advected particle positions makes the Lagrangian view applicable.
As is common for SPH discretizations, we will elect the Lagrangian view since it holds additional desirable properties such as making conservation of mass trivial to implement and enables solving the Navier-Stokes equations for primitive quantities instead of flux quantities that may cause drift instead of oscillations due to numerical inaccuracy. We state all following equations in the Lagrangian, nonconservation form.

2.2 The Continuity Equation

Using the Lagrangian view of continuum mechanics, we can apply laws of conservation to derive equations that express invariants of each fluid element with respect to time, which is an important step towards describing the dynamics of the system as time evolves. One such equation is the **CONTINUITY EQUATION**, which expresses conservation of mass:
Consider an infinitesimally small volume element $\delta\mathcal{V}$ with density ρ. The mass of the volume δm is simply[1]:

$$\delta m = \rho \delta\mathcal{V} \tag{2.2}$$

and is invariant under the material derivative in the Lagrangian reference frame[1]:

$$\begin{aligned}
\frac{D\delta m}{Dt} &= 0 && \textit{conservation of mass} && (2.3)\\
&= \frac{D\rho\delta\mathcal{V}}{Dt} && \textit{identity 2.2} && (2.4)\\
&= \delta\mathcal{V}\frac{D\rho}{Dt} + \rho\frac{D\delta\mathcal{V}}{Dt} && \textit{product rule of calculus} && (2.5)\\
&= \frac{D\rho}{Dt} + \rho\left(\frac{1}{\delta\mathcal{V}}\frac{D\delta\mathcal{V}}{Dt}\right) && \textit{divide by } \delta\mathcal{V} && (2.6)
\end{aligned}$$

We can now apply the **DIVERGENCE THEOREM** to relate $\frac{D\mathcal{V}}{Dt}$ to the divergence of the velocity across the volume of the element, where $\partial\mathcal{V}$ is its surface and $\vec{n}$ the corresponding unit normal vector[1]:

$$\frac{D\mathcal{V}}{Dt} = \oint_{\partial\mathcal{V}} \vec{v} \cdot \vec{n} \, dS = \int_{\mathcal{V}} (\nabla \cdot \vec{v}) \, d\mathcal{V} \tag{2.7}$$

As the volume $\mathcal{V}$ approaches the infinitesimal volume element $\delta\mathcal{V}$ of interest, the velocity in the volume becomes constant, the integral vanishes, and it holds that[1]:

$$\frac{D(\delta\mathcal{V})}{Dt} = (\nabla \cdot \vec{v})\,\delta\mathcal{V} \tag{2.8}$$

Substituting Equation 2.8 into Equation 2.6 we finally obtain the continuity equation:

$$\boxed{\frac{D\rho}{Dt} + \rho\,(\nabla \cdot \vec{\boldsymbol{v}}) = 0} \tag{2.9}$$

This is one of the Navier-Stokes equations in its derivative form, as opposed to the more general integral form[1]. When we additionally assume that the fluid is incompressible across a wide range of pressures, as is often done when simulating hydrodynamics, we can assert that the density of the fluid element in a Lagrangian reference frame is constant, meaning $\frac{D\rho}{Dt} = 0$ and therefore the velocity field of the flow for constant density is divergence-free[3]:

$$\nabla \cdot \vec{\boldsymbol{v}} = 0 \tag{2.10}$$

In the following sections, the fluid will generally be assumed to be incompressible.

An alternative derivation of the continuity equation uses the **Reynolds Transport Theorem**, which describes the material derivative of a scalar or tensor quantity $q(\vec{\boldsymbol{x}}, t)$ integrated over a volume as the sum of its time rate of change within the volume and the flux of the quantity through the volume's surface[3]:

$$\frac{D}{Dt}\int_{\mathcal{V}} q(\vec{\boldsymbol{x}}, t)\,dV = \int_{\mathcal{V}} \frac{\partial q(\vec{\boldsymbol{x}}, t)}{\partial t}\,dV + \oint_{\partial\mathcal{V}} q(\vec{\boldsymbol{x}}, t)(\vec{\boldsymbol{v}} \cdot \vec{\boldsymbol{n}})\,dS \tag{2.11}$$

This derivation goes as follows[3]:

$$0 = \frac{D}{Dt}\int_{\mathcal{V}} \rho\,dV \qquad \textit{conservation of mass} \tag{2.12}$$

$$= \int_{\mathcal{V}} \frac{\partial\rho}{\partial t}\,dV + \oint_{\partial\mathcal{V}} \rho(\vec{\boldsymbol{v}} \cdot \vec{\boldsymbol{n}})\,dS \qquad \textit{Reynolds Transport Theorem} \tag{2.13}$$

$$= \int_{\mathcal{V}} \frac{\partial\rho}{\partial t}\,dV + \int_{\mathcal{V}} \nabla \cdot (\rho\vec{\boldsymbol{v}})\,dV \qquad \textit{Divergence Theorem} \tag{2.14}$$

$$= \int_{\mathcal{V}} \left(\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\vec{\boldsymbol{v}})\right) dV \qquad \textit{combine integrals} \tag{2.15}$$

$$= \int_{\mathcal{V}} \left(\frac{D\rho}{Dt} + \rho\nabla \cdot \vec{\boldsymbol{v}}\right) dV \qquad \textit{constant density, Lagrangian framework} \tag{2.16}$$

$$\overset{\forall\mathcal{V}}{\Longrightarrow} \frac{D\rho}{Dt} + \rho\,(\nabla \cdot \vec{\boldsymbol{v}}) = 0 \qquad \textit{integral holds for all } \mathcal{V} \tag{2.17}$$

This use of the Reynolds Transport Theorem is very similar to the derivation that follows in section 2.3, which is why this alternative formulation was stated.

2.3 The Cauchy Momentum Equation

Mass is not the only conserved quantity that can be formulated in terms of a volume integral which can be transformed into a more convenient form using Reynolds Transport Theorem: a vital step in the derivation of the Navier-Stokes equations comes from applying the same concept to the conservation of momentum. In fact, the **Cauchy momentum equation**, which is the general case of the more specific momentum equation used in the Navier-Stokes equations, can be derived similarly to section 2.2, additionally using the continuity equation itself and Newton's second law.

We begin by observing that the change of momentum of a fluid volume $\mathcal{V}$ can be defined as the material derivative of the momentum $\int_{\mathcal{V}}(\rho\vec{\boldsymbol{v}})\,dV$ and simplify the resultant expression[3]:

$$\frac{D}{Dt}\int_{\mathcal{V}}(\rho\vec{\boldsymbol{v}})\,dV \qquad \textit{define change in momentum} \tag{2.18}$$

$$= \int_{\mathcal{V}} \frac{\partial(\rho\vec{\boldsymbol{v}})}{\partial t}\,dV + \oint_{\partial\mathcal{V}} \rho\vec{\boldsymbol{v}}(\vec{\boldsymbol{v}}\cdot\vec{\boldsymbol{n}})\,dS \qquad \textit{Reynolds Transport Theorem 2.11} \tag{2.19}$$

$$= \int_{\mathcal{V}} \frac{D}{Dt}(\rho\vec{\boldsymbol{v}})\,dV + \int_{\mathcal{V}}(\rho\vec{\boldsymbol{v}})\nabla\cdot\vec{\boldsymbol{v}}\,dV \qquad \textit{Divergence Theorem} \tag{2.20}$$

$$= \int_{\mathcal{V}} \rho\frac{D\vec{\boldsymbol{v}}}{Dt} + \vec{\boldsymbol{v}}\frac{D\rho}{Dt} + (\rho\vec{\boldsymbol{v}})\nabla\cdot\vec{\boldsymbol{v}}\,dV \qquad \textit{product rule on first integral} \tag{2.21}$$

$$= \int_{\mathcal{V}} \rho\frac{D\vec{\boldsymbol{v}}}{Dt} + \vec{\boldsymbol{v}}\underbrace{\left(\frac{D\rho}{Dt} + \rho\nabla\cdot\vec{\boldsymbol{v}}\right)}_{\text{continuity equation}=0}\,dV \qquad \textit{factor out } \vec{\boldsymbol{v}} \tag{2.22}$$

$$= \int_{\mathcal{V}} \rho\frac{D\vec{\boldsymbol{v}}}{Dt}\,dV \tag{2.23}$$

Then, we use Newton's second law, best known in its form $F = m\vec{\boldsymbol{a}}$, to assert that this change in momentum $m\vec{\boldsymbol{a}}$ is equal to the sum of forces exerted on the fluid volume, which can be decomposed into body forces $\vec{\boldsymbol{b}}^{ext}$ per unit mass[3] that act on the entire fluid mass homogeneously 'at a distance'[1], like gravity for example, and into surface forces described by stress vectors $\vec{\boldsymbol{t}}$ integrated over the fluid element's surface[3]:

$$\int_{\mathcal{V}} \rho\frac{D\vec{\boldsymbol{v}}}{Dt}\,dV = \oint_{\partial\mathcal{V}} \vec{\boldsymbol{t}}\,dS + \rho\vec{\boldsymbol{b}}^{ext} \tag{2.24}$$

One can define the **Cauchy stress tensor** $\mathbb{T}$ (sometimes referred to as σ) for the material such that it satisfies $\mathbb{T}\vec{\boldsymbol{n}} = \vec{\boldsymbol{t}}$[3]. Then, the divergence theorem may be applied again and the total forces acting on the fluid element written as:

$$\int_{\mathcal{V}} \nabla\cdot\mathbb{T}\,dV + \rho\vec{\boldsymbol{b}}^{ext} \tag{2.25}$$

Setting the expressions for total force in Equation 2.25 and total change of momentum in Equation 2.23 equal according to Newton's Law, we obtain:

$$\int_{\mathcal{V}} \rho\frac{D\vec{\boldsymbol{v}}}{Dt} - \nabla\cdot\mathbb{T} - \rho\vec{\boldsymbol{b}}^{ext}\,dV = 0 \tag{2.26}$$

From this, we have obtained the **Cauchy Momentum Equation** as our equation of motion[2]:

$$\boxed{\rho\frac{D\vec{\boldsymbol{v}}}{Dt} = \nabla\cdot\mathbb{T} + \rho\vec{\boldsymbol{b}}^{ext}} \tag{2.27}$$

2.4 The Lagrangian Navier-Stokes Equations

With the Cauchy momentum equation we have reached the end of what can be modelled using general physical principles and continuum mechanics and is valid for a range of materials. To close the system of equations for fluid flow, generality must be given up and specific assumptions about the behaviour of fluids must be used to model the specific stress tensor $\mathbb{T}$ representing incompressible, linearly viscous or Newtonian fluids. In order to derive the form of the tensor, we make the further assumptions about the fluid that will later be clarified:

1. Fluids cannot sustain shear stresses when in rigid body motion.
2. Viscosity depends on the symmetric component of the gradient of velocity, it is linearly proportional to the rate of deformation tensor.

All remaining terms of the Cauchy momentum equation are clear, only the stress tensor $\mathbb{T}$ needs to be elaborated upon. First, it can be noted that $\mathbb{T}$ is a linear transformation[3] and that the tensor is symmetric[3], as in equal to its transpose $\mathbb{T}^T = \mathbb{T}$ or $\mathbb{T}_{ij} = \mathbb{T}_{ji}$. This means that in three dimensions for example, only six degrees of freedom actually exist in this tensor[4].

The element $\mathbb{T}_{ij}$ expresses a stress along some axis $\vec{e_i}$ acting on a plane perpendicular to $\vec{e_j}$, which means that the diagonal elements $\mathbb{T}_{ii}$ are normal stresses called *tensile stresses* for negative values and *compressive stresses* for positive values of $\mathbb{T}_{ii}$[3], while $\forall i \neq j : \mathbb{T}_{ij}$ refer to *shear stresses*[1].

To make this tensor more tractable, it can be assumed that a fluid is a material which cannot sustain shear stresses when in rigid body motion, including rest[3] (assumption 1) - this means that when in rigid body motion, the stress vector on any plane is normal to that plane[3], the stress is therefore isotropic and $\mathbb{T}$ must be represented by the only isotropic second order tensor $\lambda\mathbb{1}$ or $\lambda\delta_{ij}$ for some $\lambda \in \mathbb{R}$ where δ_{ij} is the Kronecker delta[5]. This motivates a decomposition of $\mathbb{T}$ for any general motion into a sum of an isotropic tensor describing *volumetric stress* caused by pressure forces and the *deviatoric stress* $\mathbb{V}$ which simply describes deviation of the total stress $\mathbb{T}$ from the volumetric stress[6]:

$$\mathbb{T} = \mathbb{V} - p\mathbb{1} \tag{2.28}$$

Conventionally, the pressure p is defined such that a positive pressure causes a negative stress, meaning the pressure acts normal to the surface and is directed into the fluid volume $\mathcal{V}$[4]. For a fluid at rest $\mathbb{V}_{ij} = 0$ holds and the normal stress is isotropically $-p$ according to *Pascal's law*[5]. Equation 2.28 decomposes stresses into a part caused by pressure and one caused by viscosity, which is why $\mathbb{V}$ is sometimes referred to as the *viscous stress tensor*[4]. Viscosity can be thought of as internal friction in a fluid or its resistance to deformation.

The remaining term $\mathbb{V}$ is caused by viscosity and modelled according to assumption 2 in terms of the gradient of the velocity. This makes intuitive sense: where the velocity is homogeneous, and the gradient is zero, there is no friction between fluid elements - where the velocity differs greatly, there is more friction. Since velocity is a vector quantity, the gradient $\nabla\vec{v}$ is a tensor[4]:

$$(\nabla\vec{v})_{ij} = \partial_j v_i = \frac{\partial v_i}{\partial x_j} \tag{2.29}$$

As always, we can decompose this tensor $\mathbb{L} := \nabla\vec{v}$ into a sum of a symmetric and an antisymmetric part[3]:

$$\mathbb{L} = \mathbb{D} + \mathbb{W} \tag{2.30}$$

$$\mathbb{D} = \frac{1}{2}\left(\mathbb{L} + \mathbb{L}^T\right) \tag{2.31}$$

$$\mathbb{W} = \frac{1}{2}\left(\mathbb{L} - \mathbb{L}^T\right) \tag{2.32}$$

$$\tag{2.33}$$

$\mathbb{D}$ is referred to as the **RATE OF DEFORMATION TENSOR** and $\mathbb{W}$ is called the **SPIN TENSOR**.

This decomposition is convenient since the spin tensor does not contribute to viscosity and only the rate of deformation tensor may be focused on. Note that since the deviatoric stress $\mathbb{V}$ we are trying to approximate is symmetric, and it only makes sense to use the symmetric component of the velocity gradient to model it.

Intuitively, the spin tensor encodes the rotational component of the velocity gradient, and a steadily rotating fluid (where $\mathbb{D} =$) is like a rigid body rotation: the relative positions of the fluid elements do not change, only their orientation with respect to a fixed reference frame, and therefore there is no friction. There is a vector $\vec{\omega}$ such that for any $\vec{v}$ it holds that $\mathbb{W}\vec{v} = \vec{\omega} \times \vec{v}$, where $\vec{\omega}$ points in the axis of rotation with a length of the angular velocity[3]. This is why the spin tensor is closely related to the vorticity tensor $2\mathbb{W}$[3]. In fact, enforcing that viscosity shall not affect the rotational component of velocity gradients and preserving accurate vorticity is key to accurately simulating turbulences in incompressible flows and conserving angular momentum[7].

Focusing further on the rate of deformation tensor, assumption 2 can now fully be appreciated. One defining characteristic of Newtonian fluids is the assumption dating back to Isaac Newton that viscosity depends *linearly* on the rate of deformation tensor[1]. This means that terms of an order higher than linear may be neglected for small velocity gradients[4] and constant terms cannot occur since shear stress is only proportional to the rate of deformation, not the state thereof[3]: if a shear stress is applied to a fluid it will eventually continuously deform at some non-zero rate but will remain in that deformed state if the stress is removed, unlike purely elastic materials[3]. In other words $\mathbb{V}$ must vanish when the velocity is homogeneous since there is no friction in that case[4].

We now know that for incompressible fluids $\mathbb{V}$ is of the form[4]:

$$\begin{aligned}
\mathbb{V} &= 2\mu\mathbb{D} + \lambda(\nabla\cdot\vec{\boldsymbol{v}})\mathbb{1} && (2.34)\\
&= \frac{2\mu}{2}\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right) + \underbrace{\lambda(\nabla\cdot\vec{\boldsymbol{v}})\mathbb{1}}_{\text{incompressibility = 0}} && (2.35)\\
&= \mu\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right) && (2.36)
\end{aligned}$$

where μ is the dynamic viscosity[1] or first-order viscosity[4]. A second-order viscosity λ exists for compressible flows[1], but can be neglected here.

Combining the deviatoric stress with the volumetric stress, the **CONSTITUTIVE RELATION** for the stress tensor $\mathbb{T}$ of an incompressible, Newtonian fluid is finally obtained[2]:

$$\boxed{\mathbb{T} = -p\mathbb{1} + \mu\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right)} \tag{2.37}$$

With the constitutive relation in hand, the Cauchy momentum equation can be revisited, and Equation 2.37 can be inserted into Equation 2.27:

$$\begin{aligned}
\rho\frac{D\vec{\boldsymbol{v}}}{Dt} &= \nabla\cdot\left(-p\mathbb{1} + \mu\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right)\right) + \rho\vec{\boldsymbol{b}}^{ext} && \textit{insert Eq. 2.37 into Eq. 2.27} && (2.38)\\
\rho\frac{D\vec{\boldsymbol{v}}}{Dt} &= \nabla\cdot(-p\mathbb{1}) + \mu\nabla\cdot\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right) + \rho\vec{\boldsymbol{b}}^{ext} && \nabla\cdot\ \textit{is linear} && (2.39)\\
\frac{D\vec{\boldsymbol{v}}}{Dt} &= -\frac{1}{\rho}\nabla p + \nu\nabla\cdot\left((\nabla\vec{\boldsymbol{v}}) + (\nabla\vec{\boldsymbol{v}})^T\right) + \vec{\boldsymbol{b}}^{ext} && \nabla\cdot(-p\mathbb{1}) = -\nabla p\textit{, divide by } \rho && (2.40)\\
\frac{D\vec{\boldsymbol{v}}}{Dt} &= -\frac{1}{\rho}\nabla p + \nu\left(\underbrace{\nabla\cdot(\nabla\vec{\boldsymbol{v}})}_{=\nabla^2\vec{\boldsymbol{v}}} + \underbrace{\nabla\cdot(\nabla\vec{\boldsymbol{v}})^T}_{=0}\right) + \vec{\boldsymbol{b}}^{ext} && \nabla\cdot\ \textit{is linear} && (2.41)\\
\frac{D\vec{\boldsymbol{v}}}{Dt} &= -\frac{1}{\rho}\nabla p + \nu\,\nabla^2\vec{\boldsymbol{v}} + \vec{\boldsymbol{b}}^{ext} && \square && (2.42)
\end{aligned}$$

A few things of note happen in this derivation:

- The kinematic viscosity ν is defined as $\frac{\mu}{\rho}$ and inserted in Equation 2.40
- The identity $\nabla\cdot(-p\mathbb{1}) = -\nabla\cdot\begin{bmatrix}p & 0 & 0\\ 0 & p & 0\\ 0 & 0 & p\end{bmatrix} = -\begin{bmatrix}\partial p/\partial x\\ \partial p/\partial y\\ \partial p/\partial y\end{bmatrix} = -\nabla p$ is used in Equation 2.40.
- For sufficiently smooth $\vec{\boldsymbol{v}}$ and $\nabla\cdot\vec{\boldsymbol{v}} = 0$ one can show using the Theorem of Schwarz that $\nabla\cdot(\nabla\vec{\boldsymbol{v}}) = \nabla^2\vec{\boldsymbol{v}}$ as annotated in Equation 2.41[4].
- Similarly, in Equation 2.41 $\nabla\cdot(\nabla\vec{\boldsymbol{v}}^T) = \nabla(\nabla\cdot\vec{\boldsymbol{v}}) = 0$ is used[4], since the continuity equation for fluids of homogeneous density implies $\nabla\cdot\vec{\boldsymbol{v}} = 0$.

With all this, the final Navier-Stokes momentum equation for incompressible Newtonian fluids in Lagrangian form is obtained in step 2.42:

$$\boxed{\frac{D\vec{\boldsymbol{v}}}{Dt} = -\frac{1}{\rho}\nabla p + \nu\,\nabla^2\vec{\boldsymbol{v}} + \vec{\boldsymbol{b}}^{ext}} \tag{2.43}$$

2.5 Equations of State

Although the momentum equation typically takes centre stage when discussing the Navier-Stokes equations, it is important to realize that the complete Navier-Stokes equations actually refer to a set of equations and the momentum equation cannot function on its own. At the very least, the continuity equation should be included, sometimes accompanied by an energy balance equation which is crucial to heat transport problems and formulates conservation of energy in viscous flows[1]. For the dynamics of the systems considered here, the continuity and momentum equations appear to be sufficient, but one field quantity remains elusive until now: We have yet to discuss how to compute pressure.

When incompressibility is strongly enforced, the continuity equation is a constraint on the momentum equation that p can be chosen to fulfil, making it a Lagrange multiplier to the equation[2]. Since strongly enforced incompressibility generally requires solving a system to solve the Poisson equation for pressure and can be more involved, a more straightforward first approach is to employ an **Equation of State** to couple pressure to known quantities. Such an equation of state can be thought of intuitively as relating strain and stress, or a deformation of a material and the potential caused by this deformation, the negative gradient of which is a force. In this case a deviation of the fluid from its rest density ρ_0 or volume V_0 causes a pressure potential, the negative gradient of which is a pressure force that counteracts the deformation, in this case compression, as demonstrated in the hydrostatic case.

There are many such equations of state to choose from. While this choice indeed encodes different physical assumption about the fluid, choice at times appears to be rather motivated by discretization and implementation details for practical reasons than general physical principles. The equation of state should be chosen with the goal of weak compressibility and well-behavedness of the system in mind and options include:

1. $p = k\rho$ or $p = k(\rho - \rho_0)$ from the ideal gas equation[8]
2. $p = k(\frac{\rho}{\rho_0} - 1)^\gamma$ from Tait's equation[8,9]
3. $p = \max\left(0, k(\rho - \rho_0)\right)$ or $p = \max\left(0, k\left(\frac{\rho}{\rho_0} - 1\right)^\gamma\right)$ to prevent negative pressure values[2]

Option 3 in particular is used to penalize relative deviations from rest density while preventing negative pressure values that may cause undesired clumping artefacts when using SPH discretizations. In the implementation used for this report, the equation:

$$p = \max\left(0, k\left(\frac{\rho}{\rho_0} - 1\right)\right) \tag{2.44}$$

was chosen by inserting $\gamma = 1$ into Tait's equation.

While the equation of state does allow the computation of the unknown pressure, it does not appear to help close the Navier-Stokes equations, since the problem was only pushed back to the seemingly arbitrary choice or some parameter k representing stiffness. It is important to note that this parameter does not however govern the magnitude of pressure per se, but only the compressibility of the fluid[2], where larger values of k yield higher incompressibility but also higher pressure accelerations and therefore demand a higher resolution of time discretization to remain numerically stable[2] and satisfy the Courant-Friedrichs-Lewy condition[10], decreasing computational efficiency. In order to ensure that the assumption of incompressibility made in the derivation of the Navier-Stokes momentum equations hold, a sufficiently large stiffness k should be chosen for a given setting, such that compressibility becomes negligible.

Approximations for the choice of k in Tait's equation such as $k \approx \frac{\rho_0 c_s}{\gamma}$ exist, where c_s is the speed of sound and relates to the speed of flow $\vec{v}_f$[8]. Other methods such as Predictive-Corrective Incompressible SPH (PCISPH for short) approximate a globally constant k specifically for SPH discretizations such that a more optimal trade-off of incompressibility and time step size might be realized. Generally, k might need to be tuned depending on the simulated scenario in a fluid solver employing an equation of state.

Smoothed Particle Hydrodynamics

In chapter 2, the governing equations of fluid flow were derived in their Lagrangian differential form for continuous field quantities. To make the simulation of fluids tractable, these equations must now be discretized in space and time so that the evolution of the system can be numerically calculated.

The temporal domain is commonly discretized into global time steps Δt that propagate the solution of the system into the future. Numerically integrating the acceleration $\vec{a}_i(t) = \frac{D\vec{v}_i(t)}{Dt}$ from the left-hand side of the momentum equation (Equation 2.43) twice with respect to time yields a change in position $\Delta\vec{x}$ that can be used to advect quantities. Symplectic Euler time integration (also referred to as semi-implicit Euler or Euler-Cromer) is very commonly used to achieve this[2]:

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t) + \Delta t\vec{a}_i(t) \tag{3.1}$$

$$\vec{x}_i(t + \Delta t) = \vec{x}_i(t) + \Delta t\vec{v}_i(t + \Delta t) \tag{3.2}$$

The subscript i in these equations indicates that quantities are evaluated at respective particle positions $\vec{x}_i$, which are advected with the velocity field. This is why the Lagrangian form is applicable, and the material derivative can be implemented as a total derivative with respect to time.

The spatial discretization of the problem is less straightforward and yields different methods depending on the scheme chosen.

3.1 Spatial Discretization of the Continuum

The discretization chosen here makes use of **Smoothed Particle Hydrodynamics** or **SPH** for short, which was devised independently by Lucy[11] as well as Gingold and Monaghan[12] in 1977. Despite its name, this scheme has little to do with Hydrodynamics per se and does not even strictly require a particle representation of quantities to work, but rather is a general framework for the interpolation of field quantities stored at discrete locations to obtain a smooth function that can be evaluated at any location.

Since the Lagrangian framework tends to favour discretizing the continuum itself over the space it exists within, regions of the continuum are here represented by so-called particles with a singular position that represent some volume or, equivalently in the incompressible case with homogeneous density, mass. It is important to keep in mind that the word *particle* in this context refers not to a physical, elementary particle or a spherical object, but rather an abstract representation of a discrete, shapeless parcel of the continuum.

SPH can be derived by considering that these particles represent a sampling of the continuous fluid domain at singular points and can be expressed as Dirac-δ distributions weighted by some quantity. The δ-distribution can be defined as a normalized:

$$\int \delta(\vec{x})\, dV = 1 \tag{3.3}$$

and obeying $\vec{x} \neq \vec{\mathbf{0}} \implies \delta(\vec{x}) = 0$. This results in a distribution that is zero everywhere but at a singularity at the origin, where a spike of undefined height shoots up and only the integral of the distribution across that spike is a well-defined function (δ itself is a *generalized function*, not one in the analytic sense[13]). The Dirac-δ can be thought of as the limit of a Gaussian distribution as the variance approaches zero and the distribution becomes ever higher and narrower[2], or as the limit of a box of fixed unit area as the width of the box approaches zero.

For this distribution representing the particles, the identity holds that for any continuous, compactly supported function $A(\vec{x})$[2]:

$$A(\vec{x}) = (A * \delta)(\vec{x}) = \int A(\vec{x}')\delta(\vec{x} - \vec{x}')dV' \tag{3.4}$$

or the convolution of A with δ is A itself. This identity can be explained from the perspective of Fourier analysis, where the δ can be defined as the constant unit function in Fourier space and therefore $\delta = \mathcal{F}^{-1}(1)$. Since the convolution theorem applies, it then holds that a convolution in the spatial domain is equivalent to a multiplication in the frequency domain and vice-versa, resulting in a multiplication by one in the case of the convolution with a δ-distribution real space, and therefore an identity.

The key insight to SPH is that the δ-distribution can be approximated by a more well-behaved function with desirable properties such as smoothness, while approximately retaining the above identity. Such a function is referred to in SPH as a **KERNEL FUNCTION** W, *smoothing kernel*[2] or *broadening function*[11], since it broadens and smooths out the Dirac-δ distribution. With this, one can then derive[2]:

$$\begin{aligned}
A(\vec{\boldsymbol{x}}) &= (A * \delta)(\vec{\boldsymbol{x}}) && \text{Equation 3.4} && (3.5)\\
&= \int A(\vec{\boldsymbol{x}}')\delta(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}')dV' && \text{Def. of convolution, \textit{sifting property} of } \delta^{13} && (3.6)\\
&\approx \int A(\vec{\boldsymbol{x}}')W(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}')dV' && \text{approximate } \delta \text{ by } W && (3.7)\\
&= \int \frac{A(\vec{\boldsymbol{x}}')}{\rho(\vec{\boldsymbol{x}}')}W(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}')\underbrace{\rho(\vec{\boldsymbol{x}}')dV'}_{=dm'} && \text{multiply by } \frac{\rho(\vec{\boldsymbol{x}}')}{\rho(\vec{\boldsymbol{x}}')} = 1 && (3.8)\\
&\approx \sum_{\vec{\boldsymbol{x}}_j \in \mathcal{S}} A_j \frac{m_j}{\rho_j} W(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}_j) && \text{approximate Integral with discrete samples} && (3.9)
\end{aligned}$$

where subscripts denote the position where a quantity is evaluated as in $A_j := A(\vec{\boldsymbol{x}}_j)$ and $\mathcal{S}$ is a set of fluid samples. This leads to the general SPH approximation for any field quantity A[2]:

$$A_i = \sum_j A_j \frac{m_j}{\rho_j} W_{ij} \tag{3.10}$$

where the sample set $\mathcal{S}$ is implicit in the notation and $W_{ij} := W(\vec{\boldsymbol{x}}_i - \vec{\boldsymbol{x}}_j)$. $\frac{m_j}{\rho_j} = V_j$ can be seen as the fluid volume that sample j represents.

Note in particular that the mass density:

$$\rho_i = \sum_j m_j W_{ij} \tag{3.11}$$

is simply a sum over kernel functions weighted by the respective mass of samples[2]. Since mass can be perfectly conserved in a Lagrangian framework, this lends itself to fluid solvers that enforce density invariance as opposed to minimizing velocity divergence and the errors of which therefore result in volume oscillations rather than loss of volume and drift - this trade-off might be desirable but is not required by the SPH scheme in general.

As briefly mentioned before, SPH simply employs the kernel function W to perform a smoothing, thereby interpolating discrete samples, and does not necessarily have to be applied only to locations that coincide with particle positions, although finding the value of field quantities at a particle position is certainly desirable in a Lagrangian fluid simulation.

Further, note that since the gradient is a linear operator it can be pulled into the sum in Equation 3.10, resulting in[2]:

$$\nabla A_i \approx \sum_j A_j \frac{m_j}{\rho_j} \nabla W_{ij} \tag{3.12}$$

such that the gradient of a field can conveniently be computed simply by evaluating the function ∇W instead of W.

3.2 Kernel Functions and Properties

So far it has been left unspecified what form exactly the kernel function W takes, although some of its required properties were alluded to. Furthermore, W is often parameterized in its support radius and smoothing length, which we will assume to be equal in the following and name h, yielding $W(\vec{\boldsymbol{x}}_{ij}, h)$, where $\vec{\boldsymbol{x}}_{ij} = \vec{\boldsymbol{x}}_i - \vec{\boldsymbol{x}}_j$. Properties of this function shall be enumerated in the following[2]:

Normalization $\int_{\mathcal{V}} W(\vec{\boldsymbol{x}}_{ij}, h) d\vec{\boldsymbol{x}}_j = 1$
is required for the approximation to be consistent.

Dirac-δ Condition $\lim_{h\to 0} W(\vec{\boldsymbol{x}}_{ij}, h) = \delta(h)$
is the motivation for the scheme in the first place and required for $A = (A * \delta) = (A * W)$ to hold in the limit.

Compact Support $\forall\, ||\vec{\boldsymbol{x}}_{ij}|| > \hbar : W(\vec{\boldsymbol{x}}_{ij}, h) = 0$
reduces the SPH sum from $\mathcal{O}(n^2)$-complexity in n particles to potentially $\mathcal{O}(n)$

Sufficient Smoothness $W \in C^n, n \geq 2$
it is desirable for the first few derivatives of W to be continuous for discretizations such as in Equation 3.12 to be viable and for second order partial differential equations to be handled with ease[2]

Positivity $\forall \vec{\boldsymbol{x}}_{ij} : W(\vec{\boldsymbol{x}}_{ij}, h) \geq 0$
while negative values of the kernel are permitted[11] and even desired in rare cases such as when modelling surface tension[14], they are typically avoided since they might yield unphysical results at suboptimal sampling for quantities that should not be negative like mass, density, volume etc.

Symmetry $W(\vec{\boldsymbol{x}}_{ij}, h) = W(\vec{\boldsymbol{x}}_{ji}, h)$
is typically desired, even if just for lack of better assumptions about the structure of the interpolated field - indeed most kernels are spherically symmetric and only depend on the distance $||\vec{\boldsymbol{x}}_{ij}||$ between two points, which is especially valid when the interpolated field is assumed to be approximately isotropic.

Note that the two required properties in this case, the normalization and δ-condition, correspond to the two approximations made in the derivation in Equation 3.7 and Equation 3.9 of the derivation of SPH: the approximation is valid as the number of samples in the kernel support goes to infinity, making the discretization of the integral exact, and as the kernel support goes to zero, making the convolution with the kernel function an exact identity[11]. It is sometimes noted that SPH can not guarantee 0-th order consistency for arbitrary samplings, however 0-th and 1st order consistency are in fact achieved if the conditions[2]:

$$\sum_j \frac{m_j}{\rho_j} W_{ij} = 1 \qquad\qquad \sum_j \frac{m_j}{\rho_j} (\vec{\boldsymbol{x}}_j - \vec{\boldsymbol{x}}_i) W_{ij} = 1 \tag{3.13}$$

hold, which can be enforced if desired by a normalization and a matrix inversion respectively[15], although this is often not required for plausible results.

Further, note that if spherical symmetry is not enforced, a kernel may be constructed that linearly interpolates quantities along the Cartesian coordinate axes, compact to not just a sphere but a box within that sphere, and that this kernel may be evaluated on a regular grid, yielding the finite difference method - if different kernels may be used at different positions, other Eulerian methods may be constructed in the same fashion. Whether SPH is therefore a generalization of grid-based methods or if the lack of the symmetry condition reduces the definition of 'SPH' to meaninglessness is a purely taxonomic question, but it underlines the expressivity of the SPH framework.

A very typical choice for a kernel function is one that is similar to a Gaussian distribution in shape but has compact support, as demanded above. There are a few intuitions as to why a Gaussian-like kernel is a very natural choice for this problem. From the perspective of signal theory, it is common[16] and natural to apply a Gaussian filter to a signal in order to smoothen it and reduce high-frequency noise, allowing for better interpolation. The Gaussian is special in the sense that it is one of the eigenfunctions of the Fourier transform, yielding a Gaussian again when transformed[17]. More specifically, an isotropic (spherically symmetric) Gaussian filter can be thought of as the optimal way to filter a signal in many regards:

- it does not overshoot when approximating step functions, being the unique function of a [18, useful class of functions] that does not create or change local extrema[16]
- it does not create new zero-crossings in the second derivative[16], which is crucial for fluid dynamics where zero-crossings of Laplacians are of interest (e.g. the pressure Poisson equation)
- it has optimal locality in space and frequency, minimizing the Heisenberg-Weyl inequality[16]. Intuitively, spatial locality has the benefit that the smoothed signal swiftly follows the original signal with minimal delay and high fidelity, while frequency locality means that the result is optimally smooth, since the Gaussian does not extend to higher frequencies and implements a stricter low-pass filter.

The last property is an interesting connection to the uncertainty principle, which is the same inequality but typically applied to momentum and locations in physics: the product of variances of a function in the spatial and frequency domains has a lower bound, which results in uncertainty when measuring in either domain, and the lowest bound is reached only by a Gaussian distribution.

Another perspective on the usefulness of the Gaussian and SPH in general is given by a probabilistic perspective on the problem. Interestingly, the original authors of SPH independently derive it from a stochastic point of view[11 12], both groups even referencing the same book on Monte Carlo techniques[19].

Suppose fluid samples representing equal masses are independently sampled from a distribution proportional to the mass density of a fluid of homogeneous density. It then holds that the mass density can be approximated by counting the number of samples within some volume $\mathcal{V}$ around a point of interest $\vec{\boldsymbol{x}}$ and normalizing the result[20]. It is equally valid to define certaincompactly supported kernel functions to weigh the samples by and sum them instead of simply counting the samples in $\mathcal{V}$ with some indicator function, potentially smoothing out the approximated field[20]. The resulting integral[12] $\rho_{est}(\vec{\boldsymbol{x}}) = \int_{\vec{\boldsymbol{x}}_j \in \mathcal{V}} W(\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}_j)\rho(\vec{\boldsymbol{x}}_j)\, d\vec{\boldsymbol{x}}_j$ can then be approximated by a Monte-Carlo estimator over discrete samples $\vec{\boldsymbol{x}}_j$ drawn from a distribution $\rho' \propto \rho$, yielding the SPH method[12]. Again, it is natural to model the kernel W to a Gaussian by invoking the central limit theorem to argue that the measured number of samples in the volume, and by proxy the estimated density, is distributed normally around the true density as this process is repeated.

A very commonly used kernel that mimics the Gaussian in shape but has compact support and is fast to evaluate by virtue of being a polynomial is the **cubic spline kernel** or M_4 Schoenberg B-spline[15 21]:

$$W(\vec{\boldsymbol{x}}, h) = \frac{\alpha}{4h^d} \begin{cases} (2-q)^3 - 4(1-q)^3 & 0 \le q < 1 \\ (2-q)^3 & 1 \le q < 2 \\ 0 & q \ge 2 \end{cases} \tag{3.14}$$

$$\nabla W(\vec{\boldsymbol{x}}, h) = \frac{\alpha}{4h^d} \frac{\vec{\boldsymbol{x}}}{||\vec{\boldsymbol{x}}||\, h} \begin{cases} -3(2-q)^2 + 12(1-q)^2 & 0 \le q < 1 \\ -3(2-q)^2 & 1 \le q < 2 \\ 0 & q \ge 2 \end{cases} \tag{3.15}$$

where $q := \frac{||\vec{\boldsymbol{x}}||}{h}$ is the distance normalized to a particle spacing h, it holds that $\hbar = 2h$, d is the number of dimensions and α is a dimensionality-dependent constant which is $\frac{2}{3}$ for 1D, $\frac{10}{7\pi}$ for 2D and $\frac{1}{\pi}$ for 3D[15]. The kernel can also be written branchlessly (and implemented as such) as[21]:

$$W(\vec{\boldsymbol{x}}, h) = \alpha \left[\max(0\,,\, 2-q)^3 - 4\max(0\,,\, 1-q)^3\right] \tag{3.16}$$

$$\nabla W(\vec{\boldsymbol{x}}, h) = \alpha \frac{\vec{\boldsymbol{x}}}{||\vec{\boldsymbol{x}}||\, h} \left[-3\max(0\,,\, 2-q)^2 + 12\max(0\,,\, 1-q)^2\right] \tag{3.17}$$

This kernel function and its first derivative are illustrated in Figure 3.1.

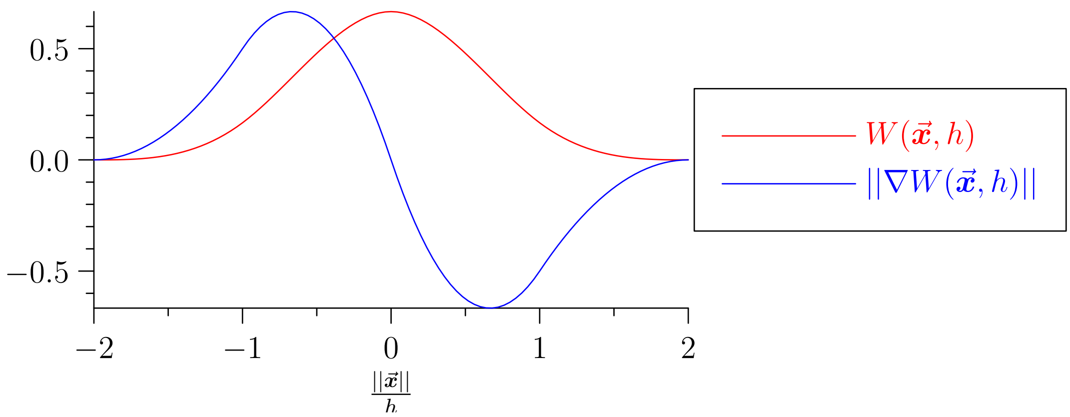


Figure 3.1: The kernel function in 1D and the magnitude of its derivative are shown with respect to the length of $\vec{x}$ normalized by the particle spacing h. It can be seen that the kernel support for this function is $\hbar = 2h$. W can also be written as parameterized by a scalar distance and is therefore spherically symmetric around the origin, while ∇W depends on the direction of $\vec{x}$ as seen in Equation 3.15 and is antisymmetric.

SOLVING THE NAVIER-STOKES EQUATIONS

Having derived the Navier-Stokes equations as the governing equations of fluid flow in chapter 2 and a method for discretizing these equations in chapter 3, we can go about actually implementing a numerical solver for the equations. Two types of solvers in particular will be discussed here: a simple solver using the equation of state to simulate weakly compressible flow and an iterative solver that uses a nearly identical formulation but can strongly enforce incompressibility. The concept of operator splitting will be key to turning a weakly compressible SPH formulation into a scheme that can be iterated to solve the pressure Poisson equation and yield incompressible flow - a task that appears rather non-trivial - using an implementation that is not much more complex than a regular state equation solver.

4.1 Equation of State SPH Solver

To construct a solver for the Navier-Stokes equation, we assume for now that an initial state and boundary conditions are given and focus on propagating the time-dependent solution into the future - details on boundaries and initialization will follow in chapter 5. Suffice it for now to know that in the approach chosen, boundaries are discretized in much the same way as the fluid and treated as tough they were fluid particles, only their positions are static.

In order to implement the kernel sum in Equation 3.10 and similar SPH sums, one needs to iterate over a set of samples denoted as subscript j. Since a kernel with compact support was chosen in Equation 3.14, all samples with non-zero contributions to the approximation of fields at $\vec{\boldsymbol{x}}_i$ lie within a radius of $\hbar$ around $\vec{\boldsymbol{x}}_i$, with $\hbar = 2h$ being a global constant since the fluid is chosen to be sampled at uniform resolution in this case. The sum $\sum_j$ therefore only has to iterate over the set of neighbouring fluid particles $\mathcal{N}_f(\vec{\boldsymbol{x}}_i) = \{\vec{\boldsymbol{x}}_j : ||\vec{\boldsymbol{x}}_i - \vec{\boldsymbol{x}}_j|| \leq \hbar\}$ and similarly for boundary particles that will be denoted with subscript k, making the computation of the sum an instance of a fixed-radius neighbourhood search problem. A uniform grid with a cell side-length of $\hbar$ is chosen to compute this set in linear time, since only a constant number of cells (9 in 2D, 27 in 3D) must be searched to find all possible neighbours of $\vec{\boldsymbol{x}}_i$. An implicit memory representation of such a grid is obtained by computing a cell index per particle, creating a list of handles which each store the particle's cell index together with its index in the attribute buffers (positions, velocities, masses, etc.), sorting the list of handles with respect to the cell index in linear time using a parallel radix sort and finally looking up all particles within some cell by searching the sorted list in logarithmic time.

With these technicalities out of the way, the discrete versions of the governing equations can be formalized. The Navier-Stokes momentum equation as stated in Equation 2.43 reads as follows, annotated for a particle of interest i:

$$\underbrace{\frac{D\vec{\boldsymbol{v}}}{Dt}}_{\text{total acceleration } \vec{\boldsymbol{a}}_i} = \underbrace{-\frac{1}{\rho}\nabla p}_{\text{pressure acceleration } \vec{\boldsymbol{a}}_i^p} + \underbrace{\nu\,\nabla^2\vec{\boldsymbol{v}}}_{\text{viscous acceleration } \vec{\boldsymbol{a}}_i^{vis}} + \underbrace{\vec{\boldsymbol{b}}^{ext}}_{\text{external accelerations } \vec{\boldsymbol{a}}_i^{ext}} \tag{4.1}$$

- Firstly, the only **EXTERNAL BODY FORCE** $\vec{\boldsymbol{b}}_i^{ext}$ per unit mass acting on the fluid is gravity, which is equal to the gravitational acceleration $\vec{\boldsymbol{g}} \approx (0, -9.81)^T$, where a 2D setting, SI units and a y-axis facing up in the positive direction are assumed in the following.
- Secondly, the **VISCOUS ACCELERATION** may be discretized. For this, an SPH approximation of the Laplacian is required, but using the less smooth and more detailed second derivative of the kernel function directly to implement this can lead to inaccurate results when sampling quality happens to be suboptimal. Instead, operating on the kernel gradient in a manner similar to a finite difference,

the following discretization can be derived[2]:

$$\nabla^2 \vec{\boldsymbol{v}} \approx 2(d+2) \sum_j \frac{m_j}{\rho_j} \frac{\vec{\boldsymbol{v}}_{ij} \cdot \vec{\boldsymbol{x}}_{ij}}{||\vec{\boldsymbol{x}}_{ij}||^2 + 0.01h^2} \nabla W_{ij} \tag{4.2}$$

where $d = 2$ is the number of dimensions and a double subscript indicates a difference as in $A_{ij} = A_j - A_i$, while the small value $0.01h^2$ is in place purely for avoiding divisions by zero and divergences if particle positions coincide[15].

From Equation 4.2 it is apparent that pairwise viscous accelerations are modelled to align with the axis spanned by the two positions of the pair and that they are symmetric, since all but the scalar quantities are projected onto $\vec{\boldsymbol{x}}_{ij}$ by virtue of the dot product and ∇W_{ij} being a scalar multiple of $\vec{\boldsymbol{x}}_{ij}$, which gives an intuition for why this formulation conserves momentum[2]. The masses m_j used in the equation are set when initializing the system and remain constant as previously mentioned, the current density ρ_j however should be calculated as outlined in Equation 3.11:

$$\rho_i \approx \sum_j m_j W_{ij} \tag{4.3}$$

- Lastly, the **PRESSURE ACCELERATION** must be discretized. For this, a symmetric formula that also conserves linear and angular momentum is chosen, since these properties are critical for robust simulations[2]:

$$-\nabla p \approx -\sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla W_{ij} \tag{4.4}$$

  To compute the pressures p_i, p_j, we use the equation of state from Equation 2.44:

$$p_i = \max\left(0, k\left(\frac{\rho_i}{\rho_0} - 1\right)\right) \tag{4.5}$$

  for some uniform rest density ρ_0 of the fluid, a stiffness parameter k that will be discussed later and the calculated densities ρ_i.

Having discretized the right-hand side of the Navier-Stokes momentum equation and weakly enforced incompressibility by linking pressure accelerations to the density deviations they are correcting through the equation of state, a procedure for calculating accelerations at some point in time t is obtained. With this, Newton's second law can be solved for the updated position $\vec{\boldsymbol{x}}_i(t+\Delta t)$ of each particle, yielding an equation of motion that is discretized in time and solved using the symplectic Euler scheme as outlined in Equation 3.1:

$$\vec{\boldsymbol{v}}_i(t+\Delta t) = \vec{\boldsymbol{v}}_i(t) + \Delta t \vec{\boldsymbol{a}}_i(t) \tag{4.6}$$

$$\vec{\boldsymbol{x}}_i(t+\Delta t) = \vec{\boldsymbol{x}}_i(t) + \Delta t \vec{\boldsymbol{v}}_i(t+\Delta t) \tag{4.7}$$

The time step Δt must be chosen to ensure a temporal resolution fine enough to resolve processes that happen on a length scale of h, which motivates the **Courant-Friedrichs-Lewy** condition or *CFL condition* for short: a particle shall not move further than its radius h in one time step, or:

$$||\vec{\boldsymbol{x}}_i(t+\Delta t) - \vec{\boldsymbol{x}}_i(t+)|| \leq \lambda h \tag{4.8}$$

where $0 < \lambda \leq 1$ describes the time step size relative to the maximum time step allowed by the condition. Since the same time step is used for all particles for simplicity, Δt must be estimated conservatively by approximating the distance the fastest particle might move in the current time step, based on the highest velocity observed in the previous time step. Further, including a maximum time step that avoids divergences when the fastest velocity is very small, such as when the simulation is initialized with zero velocities, the formulation used in this report is:

$$\Delta t = \min\left(\Delta t_{max}\,,\, \lambda \frac{h}{\max_i ||\vec{\boldsymbol{v}}_i||}\right) \tag{4.9}$$

This completes the simulation loop, allowing solutions to be propagated through time and solving the dynamics of the system. A summary of the algorithm is given in algorithm 1.

Algorithm 1 Equation of State SPH Fluid Solver *EOSSPH*

function EOSSPH($\langle\vec{\boldsymbol{x}}_i(t)\rangle, \langle\vec{\boldsymbol{v}}_i(t)\rangle, \langle m_i\rangle, \vec{\boldsymbol{g}}, \nu, k, \lambda, \rho_0$)

Step 1 – Fixed Radius Neighbour Search

$\mathcal{N}_f(\vec{\boldsymbol{x}}_i) \leftarrow \{\vec{\boldsymbol{x}}_j : ||\vec{\boldsymbol{x}}_i - \vec{\boldsymbol{x}}_j|| \leq \hbar\}$

Step 2 – Iteration: compute quantities with dependency on inputs

$\rho_i \leftarrow \sum_j m_j W_{ij}$ ▷ update density Equation 3.11

$a_i^{ext} \leftarrow \vec{\boldsymbol{g}}$ ▷ external body forces

Step 3 – Iteration: compute quantities with dependency on $\langle\rho_i\rangle$

$a_i^{vis} \leftarrow 2\nu(d+2)\sum_j \frac{m_j}{\rho_j}\frac{\vec{\boldsymbol{v}}_{ij}\cdot\vec{\boldsymbol{x}}_{ij}}{||\vec{\boldsymbol{x}}_{ij}||^2+0.01h^2}\nabla W_{ij}$ ▷ viscous acceleration Equation 4.2

$p_i \leftarrow \max\left(0, k\left(\frac{\rho_i}{\rho_0}-1\right)\right)$ ▷ pressure update Equation 2.44

$a_i^p \leftarrow -\sum_j \frac{m_j}{\rho_j}\left(\frac{p_i}{\rho_i^2}+\frac{p_j}{\rho_j^2}\right)\nabla W_{ij}$ ▷ pressure acceleration Equation 4.4

Step 4 – Numerical Time Integration

$\Delta t \leftarrow \min\left(\Delta t_{max}\, , \, \lambda\frac{h}{\max_i||\vec{\boldsymbol{v}}_i||}\right)$ ▷ CFL condition Equation 4.9

$\vec{\boldsymbol{v}}_i(t+\Delta t) \leftarrow \vec{\boldsymbol{v}}_i(t) + \Delta t\left(a_i^{ext}+a_i^{vis}+a_i^p\right)$ ▷ explicit velocity update Equation 3.1

$\vec{\boldsymbol{x}}_i(t+\Delta t) \leftarrow \vec{\boldsymbol{x}}_i(t) + \Delta t\vec{\boldsymbol{v}}_i(t+\Delta t)$ ▷ implicit position update Equation 3.1

return $\langle\vec{\boldsymbol{x}}_i(t+\Delta t)\rangle\, \langle\vec{\boldsymbol{v}}_i(t+\Delta t)\rangle$

end function

4.2 Incompressible SPH Solver

In order to move towards an incompressible solver, the update to the velocity in algorithm 1 must be inspected more closely. A main focus of fluid solvers that simulate hydrodynamics is the pressure force, since viscosity is not typically dominant and pressure forces are critical to ensure incompressibility. These pressure forces are functions of the current position of all particles, which yield some density and pressure - the more accurate the estimated positions are, the more accurately can the pressure force be calculated. Further, note how the update to the velocity integrates a sum of accelerations to a sum of velocities that are added to the current velocity, with one component of the sum being caused by pressure and two by non-pressure forces. One can equivalently compute these parts of the sum in sequence, as in:

$$\vec{\boldsymbol{v}}_i(t)' \leftarrow \vec{\boldsymbol{v}}_i(t) + \Delta t\left(a_i^{ext}+a_i^{vis}\right) \tag{4.10}$$

$$\vec{\boldsymbol{v}}_i(t+\Delta t) \leftarrow \vec{\boldsymbol{v}}_i(t)' + \Delta t a_i^p \tag{4.11}$$

So far, nothing is lost and nothing is gained through this transformation, but things change if the pressure can be formulated in direct functional dependence of the current velocity: by the time that pressure accelerations are computed, a better estimate for the velocity $\vec{\boldsymbol{v}}_i(t+\Delta t)$ is available in the form of $\vec{\boldsymbol{v}}_i(t)'$, which incorporates how gravity and viscous forces will affect the particle movement in the current time step. In order to make use of this additional knowledge, positions could be updated to yield $\vec{\boldsymbol{x}}_i'(t)$, a neighbour search could be executed and updated densities and pressure could be found as a result, however this is immensely inefficient, since the neighbour search already occupies a significant, if not dominant, part of the computation time of each simulation step.

Instead, an approximation can be applied at the current position that estimates a predicted density ρ_i^* using the updated velocities without actually advecting the particles yet. The velocities are used to approximate the time rate of change of density due to the viscous and external forces $\frac{D\rho_i}{Dt}$, which can then be multiplied by the time step size Δt to obtain a change in density $\Delta\rho$[21]:

$$\rho_i^* = \rho_i + \Delta\rho \tag{4.12}$$

$$= \rho_i + \Delta t\frac{D}{Dt}\rho_i \tag{4.13}$$

$$= \sum_j m_j W_{ij} + \Delta t\sum_j m_j(\vec{\boldsymbol{v}}_i^* - \vec{\boldsymbol{v}}_j^*)\cdot\nabla W_{ij} \tag{4.14}$$

The second term can also be interpreted as a change in density caused by velocity divergence which is caused by the viscous and external accelerations - it can be derived by applying one possible SPH discretization of the divergence operator for a vector quantity $\vec{\boldsymbol{a}}$[22]:

$$\nabla \cdot \vec{\boldsymbol{a}}_i = -\frac{1}{\rho} \sum_j m_j \vec{\boldsymbol{a}}_{ij} \cdot \nabla W_{ij} \tag{4.15}$$

to the $\frac{D\rho}{Dt}$ term in the continuity equation as seen in 2.9:

$$\begin{aligned}
\frac{D\rho_i}{Dt} &= -\rho_i(\nabla \cdot \vec{\boldsymbol{v}}_i) && \textit{Continuity, Equation 2.9} && (4.16)\\
&\approx -\rho_i \left(-\frac{1}{\rho_i} \sum_j m_j \vec{\boldsymbol{v}}_{ij} \cdot \nabla W_{ij} \right) && \textit{SPH divergence, Equation 4.15} && (4.17)\\
&= \sum_j m_j \vec{\boldsymbol{v}}_{ij} \cdot \nabla W_{ij} && \textit{simplify} && (4.18)
\end{aligned}$$

With this approximation, the updated velocity $\vec{\boldsymbol{v}}^*$ can be used, resulting in a technique referred to as **OPERATOR SPLITTING**. The original partial differential equations are split up into a sequence of sub-problems: one that solves for accelerations caused by non-pressure forces and second one that uses the result of the first problem to solve for pressure accelerations which attempt to enforce incompressibility[2]. The first problem updates velocities to $\vec{\boldsymbol{v}}_i^*$ in an explicit manner, while the second problem performs a somewhat 'implicit' update using the previously obtained results, in the hopes of improving stability[2]. This may be thought of as akin to the semi-implicit Euler update for time integration, in which an explicit velocity update allows for an implicit update to positions, in hopes of improving stability and accuracy.

Chapter 5

Boundary and Initial Conditions

5.1 Non-Uniform Single Layer Boundaries

5.2 Jittered Initialization and Lattices

5.3 Solving for Equilibrated Density

Chapter 6

Analysis

6.1 Oscillation Frequency and Error as a Function of Speed of Sound

6.2 Stability as a Function of Viscosity, Stiffness and Timestep

6.3 Stability over Viscosity and Stiffness

Chapter 7

Conclusion

Bibliography

[1]J. D. Anderson, *Computational Fluid Dynamics: The Basics with Applications.* McGraw-Hill series in mechanical engineering (New York, NY, May 1995).

[2]D. Koschier, J. Bender, B. Solenthaler, and M. Teschner, "Smoothed Particle Hydrodynamics Techniques for the Physics Based Simulation of Fluids and Solids", Eurographics 2019 - Tutorials (2019).

[3]W. M. Lai, D. H. Rubin, D. Rubin, and E. Krempl, *Introduction to continuum mechanics*, Third (Butterworth-Heinemann, 1999).

[4]V. John, *Finite Element Methods for Incompressible Flow Problems*, Vol. 51, Springer Series in Computational Mathematics 51 (Springer, Jan. 2016).

[5]J. Pedlosky, *12.800: Fluid Dynamics of the Atmosphere and Ocean (Course Notes)*, July 2014.

[6]N. Coleman, "A Derivation of the Navier-Stokes Equations", Ball State Undergraduate Mathematics Exchange **7** (2010).

[7]J. Bender, D. Koschier, T. Kugelstadt, and M. Weiler, "Turbulent Micropolar SPH Fluids with Foam", IEEE Transactions on Visualization and Computer Graphics **25**, 2284–2295 (2019).

[8]M. Becker and M. Teschner, "Weakly compressible SPH for free surface flows", in Proceedings of the 2007 acm siggraph/eurographics symposium on computer animation, SCA '07 (2007), pp. 209–217.

[9]J. Monaghan, "Simulating Free Surface Flows with SPH", Journal of Computational Physics **110**, 399–406 (1994).

[10]B. Solenthaler and R. Pajarola, "Predictive-corrective incompressible SPH", in Acm siggraph 2009 papers, SIGGRAPH '09 (2009).

[11]L. B. Lucy, "A numerical approach to the testing of the fission hypothesis.", The Astronomical Journal **82**, 1013–1024 (1977).

[12]R. A. Gingold and J. J. Monaghan, "Smoothed particle hydrodynamics: theory and application to non-spherical stars.", Monthly Notices of the Royal Astronomical Society **181**, 375–389 (1977).

[13]A. Falaschi, *Signal Processing and Information Theory*, 1st ed. (July 2022).

[14]N. Akinci, G. Akinci, and M. Teschner, "Versatile surface tension and adhesion for SPH fluids", ACM Trans. Graph. **32** (2013).

[15]D. J. Price, "Smoothed particle hydrodynamics and magnetohydrodynamics", Journal of Computational Physics **231**, 759–794 (2012).

[16]P. Getreuer, "A Survey of Gaussian Convolution Algorithms", Image Processing On Line **3**, 286–310 (2013).

[17]P. P. Vaidyanathan, "Eigenfunctions of the Fourier Transform", IETE Journal of Education **49**, 51–58 (2008).

[18]J. Babaud, A. Witkin, M. Baudin, and R. Duda, "Uniqueness of the Gaussian Kernel for Scale-Space Filtering", Pattern Analysis and Machine Intelligence, IEEE Transactions on **8**, 26–33 (1986).

[19]J. M. Hammersley and D. C. Handscomb, *Monte Carlo methods* (Methuen, London, 1964).

[20]E. Parzen, "On Estimation of a Probability Density Function and Mode", The Annals of Mathematical Statistics **33**, 1065–1076 (1962).

[21]M. Teschner, *Simulation in computer graphics: particle fluids*, (lecture slides), July 2024.

[22]M. Ihmsen, J. Orthmann, B. Solenthaler, A. Kolb, and M. Teschner, "SPH Fluids in Computer Graphics", Eurographics 2014 STAR - State of The Art Report (2014).