

SPHM: a MATLAB package for Smoothed Particle Hydrodynamics simulations

Marco Sutti *

September 13, 2022

Abstract

We present a MATLAB code that implements the Smoothed Particle Hydrodynamics (SPH) method. The paper reviews the continuous Navier–Stokes equations as well as their SPH approximation, adopting a coherent notation that allows to make easy reference to the code. The MATLAB implementation was heavily inspired by the earlier FORTRAN code of G. R. Liu and M. B. Liu, 2003. The code can be used for simple computational fluid dynamics simulations. Two classical benchmark problems are used to validate the algorithm: a one-dimensional shock tube and a two-dimensional shear cavity problem.

Key words. Smoothed Particle Hydrodynamics, Navier–Stokes equations, numerical approximation, boundary particles, MATLAB

Contents

1	Introduction	3
1.1	Notation	3
1.2	Structure of the code and function hierarchy	3
1.3	Outline	3
2	SPH fundamentals	7
2.1	Integral representation of a function	7
2.2	SPH approximation of the value of a function	8

*Mathematics Division, National Center for Theoretical Sciences, Taipei, Taiwan (msutti@ncts.tw).

3	Navier–Stokes equations	8
3.1	Continuity equation	9
3.2	Constitutive equations for fluids	9
3.2.1	Navier–Poisson law of compressible viscous fluids	10
3.2.2	Compressible viscous fluids with no bulk viscosity	10
3.2.3	Incompressible viscous fluids	11
3.3	Equations of motion	11
3.3.1	Compressible fluids with bulk viscosity	11
3.3.2	Compressible fluids with no bulk viscosity	12
3.3.3	Incompressible flows	12
3.3.4	Incompressible versus compressible flow analysis	12
3.4	Energy equation	13
3.5	Equation of state	14
3.5.1	Equation of state for gas	14
3.5.2	Equation of state for artificial water	14
4	SPH approximation of Navier–Stokes equation	15
4.1	SPH approximation of density	16
4.1.1	Summation density	16
4.1.2	Continuity density	16
4.2	SPH approximation of the equations of motion	16
4.3	SPH approximation of the deviator	17
4.4	SPH approximation of energy	18
5	Other numerical aspects	19
5.1	Artificial viscosity	19
5.2	Artificial heat	20
5.3	Variable smoothing length	20
5.4	Time step	21
5.4.1	Adaptive time step	22
6	Boundary treatment	22
6.1	Ghost particles	24
6.1.1	Pros and contras	26
6.2	Lennard–Jones potential	26
6.2.1	Pros and contras	27
6.3	Boundary force approach	28
6.3.1	Normal force	28
6.3.2	Choice of the function $B(x, y)$	29
6.3.3	Pros and contras	31
6.3.4	Radial force	32

7	Simulations	33
7.1	Sod shock tube problem	33
7.2	Shear-driven cavity problem	35
8	Floating objects	36
8.1	Equations of motion	36

1 Introduction

The particle method is not only an approximation of the continuum fluid equations, but also gives the rigorous equations for a particle system which approximates the molecular system underlying the continuum equations.

— Von Neumann, 1944

This paper introduces the smoothed particle hydrodynamics (SPH) method, a mesh-free particle method for fluid dynamics simulations. Its purpose is to serve as a documentation to the SPHM package, which can be downloaded from <https://github.com/MarcoSutti/SPHM>. SPH has undergone several improvements and has been declined in many variants over the years, and this paper does not mean to be an exhaustive introduction or review to SPH, for which we refer the reader to the extensive review papers [Mon05, Mon12].

This document is designed for scholars who are familiar with the continuous form of the Navier–Stokes equations, and wish to get a quick understanding of how to use the SPH method to obtain their approximated counterparts.

1.1 Notation

We list here the notations and symbols adopted, in order of appearance in the paper. Symbols that are only used in one section are typically omitted from the list. Some symbols are inevitably overloaded, but their meaning should be clear from the context.

1.2 Structure of the code and function hierarchy

Figure 1 illustrates the function hierarchy of SPHM.

1.3 Outline

The remaining part of this document is organized as follows. Section 2 introduces the fundamentals of SPH and of the SPH discretization technique. Section 3 gives an overview

f	Real-valued function of a real variable
Ω	Domain of SPH simulation
\mathbb{R}	Real numbers
d	Dimension of the problem
\mathbf{x}, \mathbf{x}'	Generic particles' positions
δ	Dirac's delta function
W	Smoothing kernel function
h	Smoothing length
∇	Gradient
$\nabla \cdot$	Divergence
a, b	Particle labels
$\mathbf{x}^a, \mathbf{x}^b$	Positions of particles a and b , respectively
m_a, m_b	Masses of particles a and b , respectively
ρ_a, ρ_b	Densities of particles a and b , respectively
$\ \cdot\ $	Euclidean norm of a vector
W_{ab}	Shortening for the smoothing kernel function $W(\ \mathbf{x}^a - \mathbf{x}^b\ , h)$
N	Number of particles in the support domain of particle a
∇_a	Gradient with respect to \mathbf{x}^a
ρ	Fluid density
\mathbf{v}	Velocity vector
V	Volume of fluid; velocity magnitude
$\nabla \mathbf{v}$	Velocity gradient
\mathbf{D}	Rate of deformation tensor
\mathbf{W}	Vorticity tensor
sym	Symmetric part of a matrix
τ	Matrix transpose
$I_{\mathbf{D}}$	First invariant of \mathbf{D}
Tr	Trace of a matrix
\mathbf{D}'	Strain deviator
\mathbf{I}	Identity matrix
skew	Skew-symmetric part of a matrix
\mathbf{T}	Stress tensor
p	Fluid pressure
λ, μ	Lamé constants
K	Bulk modulus
\mathbf{T}'	Shear stress tensor
\mathbf{b}	Body force
∇^2	Laplacian
Δp	Pressure difference; spacing between the boundary particles
e	Specific internal energy

T	Temperature
k_e, p_e	Kinetic and potential energy
g	Gravitational acceleration
z	Elevation
$_{,j}$	Partial derivation with respect to x_j
R	Universal gas constant
γ	Ratio of specific heats
c_s	Speed of sound
\boldsymbol{v}^{ba}	Relative velocity vector of two particles a and b
\boldsymbol{D}'^a	Rate of deformation tensor at particle a
δ_{ij}	Kronecker delta
$_{,t}$	Partial derivation with respect to time t
\boldsymbol{x}^{ba}	Relative position of two fluid particles a and b
h_0	Initial smoothing length
h_a	Smoothing length of particle a at a certain time instant
Δt	Time step
κ	Coefficient determining the size of the support domain
Δx	Initial spacing between the fluid particles
$B(a)$	Set of boundary particles in the support domain of particle a
\boldsymbol{f}_{ak}	Force per unit mass on fluid particle a due to boundary particle k
\boldsymbol{x}^{ak}	Relative position of fluid particle a and boundary particle k
\boldsymbol{f}_k	Force per unit mass on boundary particle k
\boldsymbol{f}_{ka}	Force per unit mass on boundary particle k due to fluid particle a
$\hat{\boldsymbol{n}}_k$	Outward unit normal to a boundary

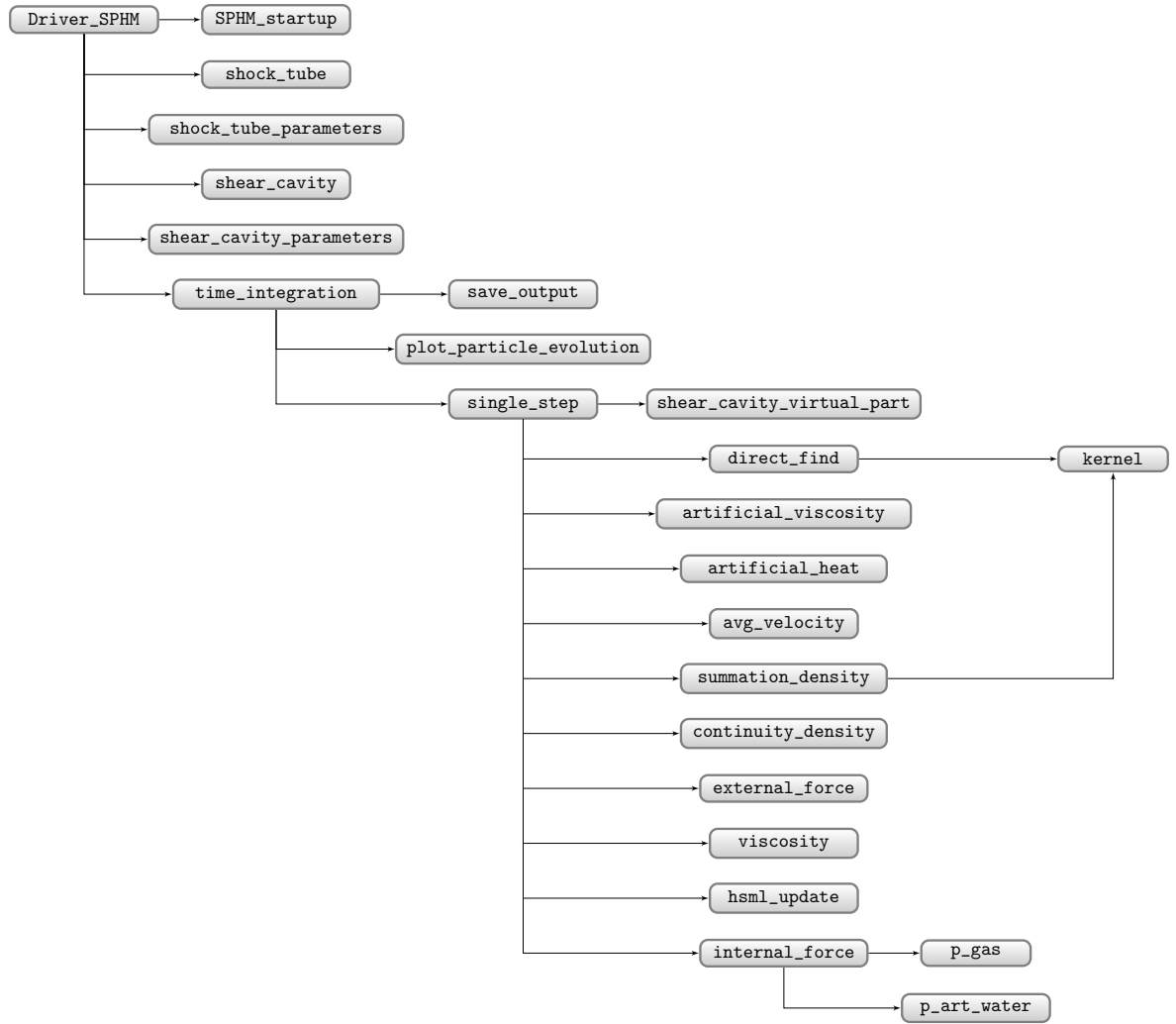


Figure 1: Function hierarchy of SPHM.

of the continuous Navier–Stokes equations. In section 4, we detail the SPH discretization of Navier–Stokes equations. Section 5 highlights some other numerical aspects we need to take care of by introducing artificial viscosity and artificial heat. In section 6, we discuss the SPH treatment of boundaries. Numerical experiments are presented in section 7. In section 8 we see how to apply SPH to floating bodies.

2 SPH fundamentals

In this section, we present a short summary of the mathematics behind the SPH discretization technique. We focus on the integral representation of a function, the SPH approximation of the value of a function, the time evolution of the smoothing length, and the time step.

2.1 Integral representation of a function

Consider a function $f: \Omega \subset \mathbb{R}^d \rightarrow \mathbb{R}$, where d is the dimension of the problem, which can be either 1, 2, or 3. The *integral representation of a function*, which can also be considered as a statement of one of the properties of Dirac’s delta function [LL03, p. 35], is

$$f(\mathbf{x}) = \int_{\Omega} f(\mathbf{x}') \delta(\mathbf{x} - \mathbf{x}') d\mathbf{x}'. \quad (1)$$

A smoothing kernel function is a function $W(\mathbf{x} - \mathbf{x}', h)$, where h is called *smoothing length* of the kernel function W , having the special property of mimicking Dirac’s delta function when h approaches zero, i.e.,

$$\lim_{h \rightarrow 0} W(\mathbf{x} - \mathbf{x}', h) = \delta(\mathbf{x} - \mathbf{x}').$$

If we replace Dirac’s delta in (1) with a smoothing kernel function $W(\mathbf{x} - \mathbf{x}', h)$ we get

$$f(\mathbf{x}) \doteq \int_{\Omega} f(\mathbf{x}') W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}'.$$

The *SPH approximation of the derivative of a function* is

$$\langle \nabla \cdot f(\mathbf{x}) \rangle = \int_{\Omega} [\nabla \cdot f(\mathbf{x}')] W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}',$$

where the symbols $\langle \ \rangle$ are used to remind us of the approximation that we are committing. In other terms, they are a reminder that the equality holds only if $h \rightarrow 0$, i.e., if $W \rightarrow \delta$.

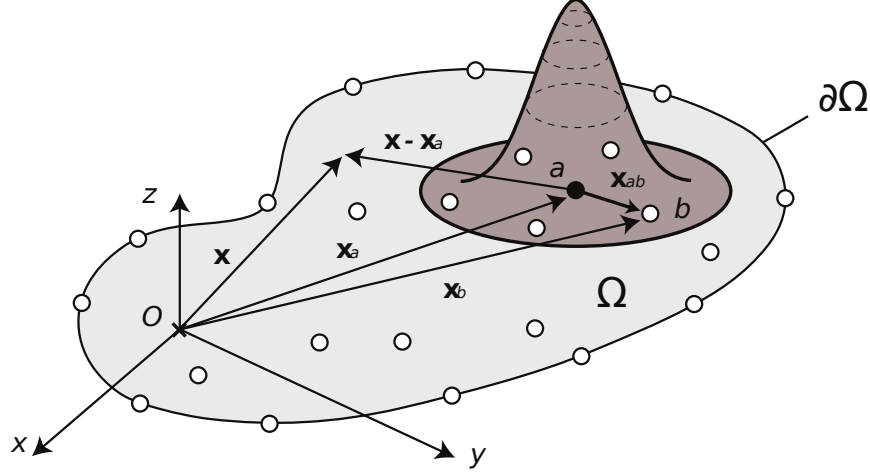


Figure 2: Illustration of SPH configuration on a two-dimensional domain.

2.2 SPH approximation of the value of a function

Let two SPH particles be labeled a and b , and let $\mathbf{x}^a, \mathbf{x}^b \in \mathbb{R}^d$ be their positions. Figure 2 depicts a two-dimensional domain Ω in the xy plane and shows a SPH kernel corresponding to particle a . The SPH kernel has a compact support, and a generic particle falling into the support domain of particle a is denoted as b .

The *SPH approximation of the value of a function at particle a* is

$$\langle f(\mathbf{x}^a) \rangle = \sum_{b=1}^N \frac{m_b}{\rho_b} f(\mathbf{x}^b) W_{ab}.$$

where N is the number of particles that fall into the support domain of particle a , m_b , ρ_b are the mass and the density of particle b , and $W_{ab} = W(\|\mathbf{x}^a - \mathbf{x}^b\|, h)$. The *SPH approximation of the value of a function derivative at particle a* is [LL03, p. 43]

$$\langle \nabla \cdot f(\mathbf{x}^a) \rangle = \sum_{b=1}^N \frac{m_b}{\rho_b} f(\mathbf{x}^b) \cdot \nabla_a W_{ab}, \quad (2)$$

where $\nabla_a W_{ab}$ indicates the gradient of W_{ab} with respect to \mathbf{x}^a .

3 Navier–Stokes equations

In this section, we review all the components of the continuous Navier–Stokes equations, namely, the continuity equation, the constitutive equations for fluids, the equations of motions and their special forms for particular types of fluids, the energy equation, and the equation of state for gas and artificial water.

The main reference for this section is [Mal77], but [RLM06] has also been used. For each equation, we typically provide the page or equation number where it can be found in these references.

3.1 Continuity equation

The continuity equation is a statement of the *principle of conservation of mass*. It reads [Mal77, eq. (5.2.7c)]

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}, \quad (3)$$

where ρ is the density, \mathbf{v} is the velocity, and $\nabla \cdot \mathbf{v}$ denotes the divergence of the velocity. Since $\nabla \cdot \mathbf{v}$ represents the time rate of change of the volume per unit volume, or *volumetric strain*, i.e.,

$$\nabla \cdot \mathbf{v} = \frac{1}{\delta V} \frac{D(\delta V)}{Dt},$$

the continuity equation (3) is telling us that the volume can change only if ρ varies in time, i.e., only if the fluid is compressible. Conversely, if the fluid is incompressible, i.e., ρ is constant, the continuity equation (3) reduces to

$$\nabla \cdot \mathbf{v} = 0. \quad (4)$$

This version of the continuity equation is called *incompressibility condition*.

3.2 Constitutive equations for fluids

In order to describe the kinematics of a fluid flow, we split the *velocity gradient* as [Mal77, eq. (4.4.5b)]

$$\nabla \mathbf{v} = \mathbf{D} + \mathbf{W},$$

where \mathbf{D} is the symmetric part of $\nabla \mathbf{v}$,

$$\mathbf{D} = \text{sym}(\nabla \mathbf{v}) = \frac{1}{2}(\nabla \mathbf{v} + \nabla \mathbf{v}^T), \quad (5)$$

and it is known as the *rate of deformation* or *stretching tensor*. Its definition is due to Euler, 1770. The so-called *first invariant* of \mathbf{D} is equal to its trace, which in turn is equal to the divergence of the velocity, i.e.,

$$I_{\mathbf{D}} = \text{Tr}(\mathbf{D}) = \nabla \cdot \mathbf{v}.$$

Moreover, the *deviator* of \mathbf{D} is defined as

$$\mathbf{D}' = \mathbf{D} - \frac{1}{3} (\text{Tr } \mathbf{D}) \mathbf{I}, \quad (6)$$

where \mathbf{I} is the identity matrix. The deviator of the rate of deformation tensor or *strain deviator* measures the change in shape of an element, since it contains the angular deformations. The term $\frac{1}{3}(\text{Tr } \mathbf{D})\mathbf{I}$ is the spherical or *hydrostatic strain*, which represents the volume change.

The *spin* or *vorticity* tensor \mathbf{W} is defined as the skew-symmetric part of the velocity gradient

$$\mathbf{W} = \text{skew}(\nabla \mathbf{v}) = \frac{1}{2}(\nabla \mathbf{v} - \nabla \mathbf{v}^\top).$$

In the following, we will review three different formulations of the constitutive equations for fluids.

3.2.1 Navier–Poisson law of compressible viscous fluids

The Navier–Poisson law of compressible viscous fluids (also known as Navier–Stokes behavior) relates the stress tensor to the rate of deformation tensor according to the following equation (see [Mal77, eq. (6.3.10)] or [RLM06, eq. (7.29)])

$$\mathbf{T} = -p\mathbf{I} + \lambda(\text{Tr } \mathbf{D})\mathbf{I} + 2\mu\mathbf{D}, \quad (7)$$

where for clarity we omit the dependence on ρ of p , λ , and μ . The first term in this relationship denotes the static component of the pressure. The second term indicates the dynamic component of the pressure, since it depends on $\text{Tr}(\mathbf{D})$. The third term represents the shear stress.

The constants λ and μ are two independent parameters characterizing the elastic behavior of a given body. They were introduced by Gabriel Lamé (1795–1870) and thus are often referred to as *Lamé constants*. They have to be determined for every material by means of experiments [Mus75, p. 59-60].

3.2.2 Compressible viscous fluids with no bulk viscosity

The *bulk modulus* (also known as *modulus of compression* or *bulk viscosity*) is defined as a function of the Lamé constants [Mal77, eq. (6.3.15)]

$$K = \lambda + \frac{2}{3}\mu.$$

The condition of zero bulk viscosity $K = 0$, also known as *Stokes condition* [Mal77, eq. (6.3.17b)], implies

$$\lambda = -\frac{2}{3}\mu. \quad (8)$$

Under this condition, (7) becomes

$$\begin{aligned} \mathbf{T} &= -p\mathbf{I} - \frac{2}{3}\mu(\text{Tr } \mathbf{D})\mathbf{I} + 2\mu\mathbf{D} \\ &= -p\mathbf{I} + 2\mu \left[\mathbf{D} - \frac{1}{3}(\text{Tr } \mathbf{D})\mathbf{I} \right]. \end{aligned}$$

Finally, recalling the definition (6), we get [Mal77, eq. (6.3.23)]

$$\mathbf{T} = \underbrace{-p\mathbf{I}}_{\text{isotropic part of } \mathbf{T}} + \underbrace{2\mu\mathbf{D}'}_{\text{deviatoric part of } \mathbf{T}}. \quad (9)$$

This equation governs the behavior of compressible viscous fluids with no bulk viscosity. The first term in the equation, namely $-p\mathbf{I}$, is the isotropic part of the stress tensor. The second term is the deviatoric part of the stress tensor, $2\mu\mathbf{D}'$, which is called *shear stress tensor* and is denoted as \mathbf{T}' [Mal77, eq. (6.3.24)]. In fact, \mathbf{T}' is clearly related to shear since \mathbf{D}' contains the angular deformations.

3.2.3 Incompressible viscous fluids

To obtain the constitutive equation for incompressible viscous fluids it is sufficient to insert in (9) the *incompressibility condition* (4). Under such condition, $\mathbf{D}' \equiv \mathbf{D}$ and therefore (9) becomes

$$\mathbf{T} = -p\mathbf{I} + 2\mu\mathbf{D}. \quad (10)$$

3.3 Equations of motion

The equations of motion for a continuous medium are derived from the *momentum principle* or *Newton's second law*, and they have the general form [RLM06, eq. (5.30)]

$$\rho \frac{d\mathbf{v}}{dt} = \nabla \cdot \mathbf{T} + \rho\mathbf{b}, \quad (11)$$

where \mathbf{b} is the body force. These equations are also known as *Cauchy's equations of motion* [RLM06]. Clearly, the specific form of the equation of motion (11) depends on the constitutive relationship for the stress tensor \mathbf{T} , as we will see in the following subsections.

3.3.1 Compressible fluids with bulk viscosity

If we specialize (11) by using the constitutive equation (7), we get [RLM06, eq. (9.71)]

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + \nabla(\lambda \nabla \cdot \mathbf{v}) + \nabla \cdot (2\mu\mathbf{D}) + \rho\mathbf{b}.$$

Recalling that, by definition (5), $2\mathbf{D} = \nabla\mathbf{v} + \nabla\mathbf{v}^T$, the previous equation can also be rewritten as

$$\begin{aligned} \rho \frac{d\mathbf{v}}{dt} &= -\nabla p + \nabla(\lambda \nabla \cdot \mathbf{v}) + \nabla \cdot (\mu \nabla \mathbf{v} + \mu \nabla \mathbf{v}^T) + \rho\mathbf{b} \\ &= -\nabla p + \nabla(\lambda \nabla \cdot \mathbf{v}) + \mu \nabla^2 \mathbf{v} + (\mu \nabla \cdot \mathbf{v}) \nabla + \rho\mathbf{b}, \end{aligned}$$

where $\nabla^2 := \nabla \cdot \nabla$ is the Laplacian. Finally we get [Mal77, eq. 7.1.14b]

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + (\lambda + \mu) \nabla(\nabla \cdot \mathbf{v}) + \mu \nabla^2 \mathbf{v} + \rho\mathbf{b}. \quad (12)$$

3.3.2 Compressible fluids with no bulk viscosity

To obtain the equations of motion for compressible fluids with no bulk viscosity it is sufficient to insert the Stokes condition (8) into (12) and get [Mal77, eq. 7.1.17b]

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + \frac{1}{3} \mu \nabla(\nabla \cdot \mathbf{v}) + \mu \nabla^2 \mathbf{v} + \rho \mathbf{b}. \quad (13)$$

This equation can also be obtained from (11) by inserting the constitutive equation for a compressible fluid with no bulk viscosity (9), i.e.,

$$\rho \frac{d\mathbf{v}}{dt} = \nabla \cdot (-p\mathbf{I} + 2\mu\mathbf{D}') + \rho \mathbf{b}.$$

This also provides an alternative expression of the equations of motion, namely,

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + \nabla \cdot \mathbf{T}' + \rho \mathbf{b}, \quad (14)$$

where $\mathbf{T}' = 2\mu\mathbf{D}'$ is the shear stress tensor.

3.3.3 Incompressible flows

It is sufficient to insert the incompressibility condition (4) into (13) and get (see [Mal77, eq. (7.1.17a)] or [RLM06, eq. (9.75)])

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \rho \mathbf{b}. \quad (15)$$

3.3.4 Incompressible versus compressible flow analysis

In this subsection, we are going to emphasize the differences between incompressible versus compressible flow analysis. Consider a three-dimensional incompressible flow at constant temperature. In such a case, we have to find four unknowns to completely describe the fluid flow, i.e., **the pressure p** and **the three components of the velocity \mathbf{v}** . As a consequence, in an incompressible flow analysis we need four equations: the incompressibility condition $\nabla \cdot \mathbf{v} = 0$ and the three scalar equations of motion from the vector equation (15). These equations are coupled since the velocity \mathbf{v} appears in both of them [cC10]. The pressure and the velocity are also coupled since they both appear in equation (15). When the velocity is known, we can integrate the pressure gradient appearing in equation (15) to compute the pressure field up to an arbitrary constant. To determine this arbitrary constant we need to measure the pressure somewhere in the field, i.e., we need a boundary condition on the pressure. Most of the codes in computational fluid dynamics do not calculate pressure by direct integration of the Navier–Stokes equations, but they use a pressure correction algorithm based on a form of *Poisson’s equation* for the pressure difference Δp [cC10]

$$\nabla^2(\Delta p) = \text{RHS}(n). \quad (16)$$

Note that since pressure appears only as a gradient in the incompressible Navier–Stokes equations (15), the absolute magnitude of the pressure is not important, only pressure differences matter. This is not true for compressible flow, where p is the thermodynamic pressure rather than the mechanical one.

To find p in compressible flows, we need to introduce an *equation of state*, which relates the pressure, temperature and density of the fluid (see section 3.5). Since the equation of state introduces *temperature* as an additional unknown, we also need an additional equation, which expresses the *conservation of energy principle* (see section 3.4). Therefore, in a compressible flow analysis, we have to deal with six equations in six unknowns, i.e., \mathbf{v} , p , e , and T . For simple compressible systems, the total energy consists of internal, kinetic and potential energy, expressed on a unit-mass basis, i.e.,

$$e = u + k_e + p_e = u + \frac{1}{2} V^2 + gz.$$

The internal energy is the sum of all microscopic forms of energy (energy related to the molecular structure of a system and the degree of molecular activity, also referred to as *thermal energy*). The kinetic energy is the energy that a system possesses as a result of its motion. The potential energy is the energy that a system possesses as a result of its elevation in a gravitational field.

3.4 Energy equation

The energy equation is the mathematical formulation of the *first law of thermodynamics*, which expresses the *conservation of energy*:

The time rate of change of the energy inside an infinitesimal fluid cell equals the summation of the heat flux into that fluid cell, and the time rate of work done by the body and surface forces acting on that fluid cell.

The general form of the energy equation, neglecting heat flux term and the body force, is ([Mal77, eq. (5.4.7)] or [RLM06, eq. (5.46)])

$$\rho \frac{de}{dt} = \mathbf{T} : \nabla \mathbf{v}. \quad (17)$$

Here, e is a specific energy per unit mass [J/kg] and $:$ denotes the scalar product of two tensors. In index notation, equation (17) can be written as

$$\rho e_{,t} = T_{ij} v_{i,j},$$

with $i, j = 1, \dots, d$.

If we consider a compressible viscous fluid with no bulk viscosity, we can use the constitutive law (9) to obtain

$$\rho \frac{de}{dt} = (-p\mathbf{I} + 2\mu\mathbf{D}') : \nabla \mathbf{v},$$

then we obtain [Mal77, eq. (7.1.14c)]

$$\rho \frac{de}{dt} = -p(\nabla \cdot \mathbf{v}) + 2\mu(\mathbf{D}' : \nabla \mathbf{v}). \quad (18)$$

The meaning of this equation is that the time rate of change of the specific internal energy is equal to the work done by the isotropic pressure multiplying the volumetric strain, plus the energy dissipation due to the viscous shear forces.

3.5 Equation of state

An equation of state is any equation that relates pressure, temperature and density of a substance [cC10]. In basic applications of SPH, we use equations of state for gas and for artificial water.

3.5.1 Equation of state for gas

The most famous equation of state is the ideal gas law [Mal77, eq. (5.6.3a)]

$$p = \rho RT,$$

where p is the absolute pressure, ρ is the density, R is the universal gas constant and T is the absolute temperature. In the SPH calculations, we use the so-called *gamma law* [Mal77, eq. (7.4.14)]

$$p = (\gamma - 1)\rho e, \quad (19)$$

with the *ratio of specific heats* $\gamma = 1.4$. As above, e denotes the specific internal energy. In SPHM, this equation of state is implemented in the script `p_gas`. Note that if the fluid is incompressible, we have no equation of state, since this is replaced by the condition that the density remains constant.

3.5.2 Equation of state for artificial water

In SPH, the water is treated as weakly compressible, thus we refer to this kind of fluid as *artificial water*. This allows the use of an equation of state to determine fluid pressure, which is much faster than solving Poisson's equation (16) as discussed in section 3.3.4. Usually, the compressibility is adjusted so that the time step of the simulation is reasonable. To approximate the real fluid as an artificial compressible fluid, a smaller value than the actual sound speed should be used so that the time step is increased to an acceptable value; but it should also be large enough so that the behavior of the artificial compressible fluid is sufficiently close to the reality [LL03].

The equation of state for artificial water most frequently used when the atmospheric pressure is negligible is [Mon05, eq. (10.1)]

$$p = B \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right],$$

where ρ_0 is the reference density, $\gamma = 7.0$ and

$$B = \rho_0 \frac{c_s^2}{\gamma} \quad (20)$$

is a parameter chosen so that the speed of sound c_s is large enough to limit the relative density variation $|\delta\rho|/\rho$. This is done to endow the fluid with a slight compressibility in order to permit the use of an equation of state. This formulation goes back as early as [Col48, p. 44].

The sound speed c_s is determined according to the *Newton–Laplace formula* for the speed of pressure waves

$$c_s = \sqrt{\frac{K}{\rho_0}}, \quad (21)$$

where K is the *bulk modulus*, which measures the substance’s resistance to uniform compression. For a gas, $K = \gamma p$, so that (21) becomes

$$c_s = \sqrt{\frac{\gamma p}{\rho_0}}.$$

Usually the sound speed c_s in water at 25°C is about 1497 m/s for freshwater and 1560 m/s for seawater.

The relative density variation is related to the *Mach number*

$$\frac{|\delta\rho|}{\rho} \approx M^2,$$

where $M = v/c_s$, with v being the maximum fluid speed. Hence we can ensure a slight density variation, say $|\delta\rho|/\rho \approx 0.01$, if $M < 0.1$, i.e., if $c_s \approx 10 v$ [MKI03]. This means that the sound speed should be about ten times faster than the maximum fluid velocity in order to keep density variations within less than 1% [GRD+10]. In fact, if we substitute $c_s = 10 v$ into (20), we get

$$B = 100 \rho_0 \frac{v^2}{\gamma},$$

and the relative density variation is ≈ 0.01 .

As can be seen, the above formulas require an estimation of the maximum fluid speed. For instance, when a dam of height H collapses, an adequate value for the maximum water velocity is $v^2 = 2gH$ [Mon94, MKI03] or $v^2 = gH$ [MK99]. The equation of state for artificial water is implemented in `p_art_water`.

4 SPH approximation of Navier–Stokes equation

In this section, we will see how to derive the SPH approximation of the Navier–Stokes equations. The boundary treatment that we will see in the next section is strictly related to the equations of motion.

4.1 SPH approximation of density

The SPH approximation of density can be performed via two different techniques: the summation density approach and the continuity density approach.

4.1.1 Summation density

The SPH approximation of density of particle a with the summation density approach is [LL03, p. 114]

$$\rho_a = \sum_{b=1}^N m_b W_{ab}, \quad (22)$$

where N is the number of particles that fall into the support domain of particle a . This is the preferred formulation for general fluid phenomena. It has the advantage of conserving the mass exactly, but it shows edge effects for particles at the interface between two materials and at the boundaries, an issue known as *boundary particle deficiency*. It is implemented in `summation_density`.

4.1.2 Continuity density

The continuity density approach is the SPH approximation of the continuity equation (3). Using (2) to approximate the velocity divergence $\nabla \cdot \mathbf{v}$ appearing in (3), one can obtain

$$\frac{d\rho_a}{dt} = -\rho_a \sum_{b=1}^N \frac{m_b}{\rho_b} v_j^{ba} \frac{\partial W_{ab}}{\partial x_j^a},$$

where the repeated index j implies summation over it, and v_j^{ba} are the components of the relative velocity \mathbf{v}^{ba} . The relative velocity vector $\mathbf{v}^{ba} = \mathbf{v}^b - \mathbf{v}^a$ has components $v_i^{ba} = v_i^b - v_i^a$, and the position vector \mathbf{x}^a of particle a has components x_i^a , for $i = 1, \dots, d$. Thanks to the introduction of the velocity difference, **this formulation takes into account the relative velocities between two particles, so that the density only changes when particles are in relative motion**. The continuity density approach is usually preferred when dealing with discontinuous phenomena, such as explosions or wave-breaking. It is implemented in `continuity_density`.

4.2 SPH approximation of the equations of motion

We recall the equations of motion for compressible fluids with no bulk viscosity (14)

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla p + \nabla \cdot \mathbf{T}' + \rho \mathbf{b}.$$

If we neglect the body forces, divide both sides by ρ , and substitute the constitutive relationship $\mathbf{T}' = 2\mu\mathbf{D}'$, we get

$$\frac{d\mathbf{v}}{dt} = -\frac{\nabla p}{\rho} + \frac{1}{\rho} \nabla \cdot (2\mu\mathbf{D}').$$

In index notation,

$$\frac{dv_i}{dt} = -\frac{1}{\rho} p_{,i} + \frac{2}{\rho} \mu D'_{ij,j},$$

with the indices $i, j = 1, 2, 3$. As usual, a repeated index implies summation over that index, and $_{,j}$ denotes partial derivation with respect to x_j .

The SPH approximation of this equation is [LL03, eq. (4.43)]

$$\frac{dv_i^a}{dt} = -\sum_{b=1}^N m_b \left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_i^a} + 2 \sum_{b=1}^N m_b \left(\frac{\mu_a D'_{ij}^a}{\rho_a^2} + \frac{\mu_b D'_{ij}^b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_j^a}. \quad (23)$$

In the function `internal_force` of SPHM, the term

$$2 \left(\frac{\mu_a D'_{ij}^a}{\rho_a^2} + \frac{\mu_b D'_{ij}^b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_j^a}$$

is stored in the variable `eom_forces`.

If we use the second form of symmetrization for the density [LL03, eq. (4.42)], the equations of motion become

$$\frac{dv_i^a}{dt} = -\sum_{b=1}^N m_b \left(\frac{p_a + p_b}{\rho_a \rho_b} \right) \frac{\partial W_{ab}}{\partial x_i^a} + 2 \sum_{b=1}^N m_b \frac{\mu_a D'_{ij}^a + \mu_b D'_{ij}^b}{\rho_a \rho_b} \frac{\partial W_{ab}}{\partial x_j^a}.$$

Here, v_i^a denotes the particle a velocity component in the i th direction. D'_{ij}^a are the coefficients of the deviator of the rate of deformation tensor for particle a .

4.3 SPH approximation of the deviator

To complete the SPH approximation of the equations of motion, we need to know how to compute the deviator of the rate of deformation tensor at particle a , \mathbf{D}'^a . We recall that the rate of deformation tensor is defined as in (6), i.e.,

$$\mathbf{D}' = \mathbf{D} - \frac{1}{3} \text{Tr}(\mathbf{D}) \mathbf{I}.$$

Moreover, $\mathbf{D} = \frac{1}{2}(\nabla \mathbf{v} + \nabla \mathbf{v}^\top)$ and $\text{Tr}(\mathbf{D}) = \nabla \cdot \mathbf{v}$, therefore

$$\mathbf{D}' = \frac{1}{2}(\nabla \mathbf{v} + \nabla \mathbf{v}^\top) - \frac{1}{3}(\nabla \cdot \mathbf{v}) \mathbf{I}.$$

In index notation, this expression reads

$$D'_{ij} = \frac{1}{2}(v_{i,j} + v_{j,i}) - \frac{1}{3}(v_{k,k}) \delta_{ij},$$

where δ_{ij} is Kronecker's delta.

Remark. When $d = 3$, the deviator \mathbf{D}' has nine components, but six of them are equal to each other because of symmetry properties, so in general one has only six independent components. Similarly, when $d = 2$, \mathbf{D}' has four components, but two of them are equal to each other because of symmetry properties, so we are left with only three independent components.

The SPH approximation for D'_{ij} is thus [LL03, eq. (4.48)]

$$D'_{ij} = \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_j^{ba} \frac{\partial W_{ab}}{\partial x_i^a} + \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_i^{ba} \frac{\partial W_{ab}}{\partial x_j^a} - \left(\frac{1}{3} \sum_{b=1}^N \frac{m_b}{\rho_b} v_k^{ba} \frac{\partial W_{ab}}{\partial x_k^a} \right) \delta_{ij}. \quad (24)$$

We expand (24) and get all the six components of \mathbf{D}'^a . The following expressions are those appearing in the SPHM function `internal_force`.

$$\begin{aligned} D'_{11} &= \sum_{b=1}^N \frac{m_b}{\rho_b} v_1^{ba} \frac{\partial W_{ab}}{\partial x_1^a} - \frac{1}{3} \sum_{b=1}^N \frac{m_b}{\rho_b} \left(v_1^{ba} \frac{\partial W_{ab}}{\partial x_1^a} + v_2^{ba} \frac{\partial W_{ab}}{\partial x_2^a} + v_3^{ba} \frac{\partial W_{ab}}{\partial x_3^a} \right), \\ D'_{22} &= \sum_{b=1}^N \frac{m_b}{\rho_b} v_2^{ba} \frac{\partial W_{ab}}{\partial x_2^a} - \frac{1}{3} \sum_{b=1}^N \frac{m_b}{\rho_b} \left(v_1^{ba} \frac{\partial W_{ab}}{\partial x_1^a} + v_2^{ba} \frac{\partial W_{ab}}{\partial x_2^a} + v_3^{ba} \frac{\partial W_{ab}}{\partial x_3^a} \right), \\ D'_{33} &= \sum_{b=1}^N \frac{m_b}{\rho_b} v_3^{ba} \frac{\partial W_{ab}}{\partial x_3^a} - \frac{1}{3} \sum_{b=1}^N \frac{m_b}{\rho_b} \left(v_1^{ba} \frac{\partial W_{ab}}{\partial x_1^a} + v_2^{ba} \frac{\partial W_{ab}}{\partial x_2^a} + v_3^{ba} \frac{\partial W_{ab}}{\partial x_3^a} \right), \\ D'_{12} &= \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_2^{ba} \frac{\partial W_{ab}}{\partial x_1^a} + \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_1^{ba} \frac{\partial W_{ab}}{\partial x_2^a}, \\ D'_{13} &= \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_3^{ba} \frac{\partial W_{ab}}{\partial x_1^a} + \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_1^{ba} \frac{\partial W_{ab}}{\partial x_3^a}, \\ D'_{23} &= \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_3^{ba} \frac{\partial W_{ab}}{\partial x_2^a} + \frac{1}{2} \sum_{b=1}^N \frac{m_b}{\rho_b} v_2^{ba} \frac{\partial W_{ab}}{\partial x_3^a}. \end{aligned}$$

4.4 SPH approximation of energy

We recall the energy equation (18) for compressible viscous fluids with no bulk viscosity

$$\rho \frac{de}{dt} = -p \nabla \cdot \mathbf{v} + 2\mu \mathbf{D}' : \nabla \mathbf{v},$$

in index notation:

$$e_{,t} = \frac{1}{\rho} \left(-p v_{k,k} + 2\mu D'_{ij} v_{i,j} \right),$$

where $i, j, k = 1, \dots, d$.

The SPH approximation of the energy equation for particle a is [LL03, p. 120]

$$\frac{de_a}{dt} = \frac{1}{2} \sum_b^N m_b \left(\frac{p_a + p_b}{\rho_a \rho_b} \right) v_j^{ab} \frac{\partial W_{ab}}{\partial x_j^a} + 2 \frac{\mu_a}{\rho_a} (D'_{ij})^2,$$

or, with the second form of symmetrization,

$$\frac{de_a}{dt} = \frac{1}{2} \sum_b^N m_b \left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2} \right) v_j^{ab} \frac{\partial W_{ab}}{\partial x_j^a} + 2 \frac{\mu_a}{\rho_a} (D'_{ij})^2.$$

where we let $(D'_{ij})^2$ denote the sum $\sum_{i,j=1}^d D'_{ij} D'_{ij}$.

We can write the extended version of the first form to obtain the expression that is implemented in SPHM, namely,

$$\begin{aligned} \frac{de_a}{dt} = & \underbrace{\frac{1}{2} \sum_b^N m_b \left(\frac{p_a + p_b}{\rho_a \rho_b} \right) \left[v_1^{ab} \frac{\partial W_{ab}}{\partial x_1^a} + v_2^{ab} \frac{\partial W_{ab}}{\partial x_2^a} + v_3^{ab} \frac{\partial W_{ab}}{\partial x_3^a} \right]}_{\text{pressure work}} \\ & + 2 \underbrace{\frac{\mu_a}{\rho_a} \left[(D'_{11})^2 + (D'_{22})^2 + (D'_{33})^2 + \underbrace{2(D'_{12})^2 + 2(D'_{23})^2 + 2(D'_{13})^2}_{\text{because of symmetry of } \mathbf{D}'} \right]}_{\text{energy dissipation due to viscous forces (viscous entropy)}}. \end{aligned}$$

In the function `internal_force` of SPHM, the term

$$(p_a + p_b) \left[v_1^{ab} \frac{\partial W_{ab}}{\partial x_1^a} + v_2^{ab} \frac{\partial W_{ab}}{\partial x_2^a} + v_3^{ab} \frac{\partial W_{ab}}{\partial x_3^a} \right]$$

is stored in the variable `p_work_part`.

This completes the presentation of the SPH approximation of Navier–Stokes equations.

5 Other numerical aspects

In this section, we present some other numerical aspects of the SPH method.

5.1 Artificial viscosity

The Monaghan type artificial viscosity was introduced in [Mon92], and reads

$$\Pi_{ab} = \begin{cases} \frac{-\alpha c_{ab} \phi_{ab} + \beta \phi_{ab}^2}{\rho_{ab}} & \text{if } \mathbf{v}^{ba} \cdot \mathbf{x}^{ba} < 0, \\ 0 & \text{if } \mathbf{v}^{ba} \cdot \mathbf{x}^{ba} \geq 0, \end{cases}$$

where \mathbf{v}^{ba} and \mathbf{x}^{ba} denote the relative velocity and the relative position, respectively, and

$$\phi_{ab} = \frac{h_{ab} \mathbf{v}^{ba} \cdot \mathbf{x}^{ba}}{\|\mathbf{x}^{ba}\|^2 + (0.1 h_{ab})^2},$$

where c_{ab} , ρ_{ab} , and h_{ab} are the average smoothing length, speed of sound, and density, respectively. The parameters α and β are usually taken equal to 1 [Mon92]. The term $(0.1 h_{ab})^2$ prevents numerical singularities when two particles are approaching each other.

As it appears from its formulation, the term Π_{ab} is nonzero only in the case of material compression ($\mathbf{v}^{ba} \cdot \mathbf{x}^{ba} < 0$). It is added to the pressure terms in the SPH approximation of the momentum and energy equations.

5.2 Artificial heat

The artificial heat was introduced in [Noh87, Ful94] with the following formulation

$$H_a = 2 \sum_{b=1}^N \frac{q_{ab}}{\rho_{ab}} \frac{e_a - e_b}{\|\mathbf{x}^{ba}\|^2 + (0.1 h_{ab})^2} \mathbf{x}^{ba} \cdot \nabla_a W_{ab},$$

where

$$\begin{aligned} q_a &= \alpha h_a \rho_a c_a |\nabla \cdot \mathbf{v}_a| + \beta h_a^2 \rho_a (\nabla \cdot \mathbf{v}_a)^2, \\ q_b &= \alpha h_b \rho_b c_b |\nabla \cdot \mathbf{v}_b| + \beta h_b^2 \rho_b (\nabla \cdot \mathbf{v}_b)^2, \end{aligned}$$

and $q_{ab} = \frac{1}{2}(q_a + q_b)$. The term H_a is added to the SPH approximation of the energy equation.

5.3 Variable smoothing length

The early implementations of SPH [GM77] adopted the same smoothing length for all the particles and kept it fixed throughout the entire simulation. Later, problems involving fluid expansion and contraction required the introduction of a smoothing length varying in space according to the local density of the particles [LL03]. Problems dealing with shocks pointed out the need of a smoothing length varying in both space and time.

Gingold and Monaghan [GM82] devised the simplest way to vary h with time so that the number of neighboring particles of a given particle a remains more or less the same. For each particle a at a certain time instant, the smoothing length h_a is given by¹

$$h_a = h_0 \left(\frac{\rho_0}{\rho_a} \right)^{1/d},$$

where h_0 is the initial smoothing length, ρ_0 is the initial density, ρ_a is the summation density at time t for particle a , see eq. (22). This proved to be a powerful way of calculating the

¹We drop the explicit dependence on t for clarity.

smoothing length since it automatically makes h varying in both space and time. The above expression has been improved by [Mon05, eq. (4.2)]

$$h_a = 1.3 \left(\frac{m_a}{\rho_a} \right)^{1/d},$$

and further modified by the same author to avoid excessively large or small values of h

$$h_a = 1.3 \left(\frac{m_a}{A + \rho_a} \right)^{1/d},$$

where A is a suitable constant. All these variants are implemented in `hsm1_update`.

Another approach proposed by [Ben90] consists in calculating h from the rate of change of density, i.e.,

$$\frac{dh}{dt} = -\frac{1}{d} \frac{h}{\rho} \frac{d\rho}{dt}. \quad (25)$$

Let us recall the continuity equation (3)

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}.$$

Substituting this expression into (25), we obtain

$$\frac{dh}{dt} = \frac{h}{d} \nabla \cdot \mathbf{v},$$

which can be discretized using SPH approximation for the velocity divergence.

5.4 Time step

To perform the time integration, SPH makes use of an explicit numerical integration scheme. Explicit methods calculate the state of a system at a later time from the state of the system at the current time. For instance, given the equation

$$Y(t + \Delta t) = F(Y(t)),$$

where Δt is the time step, F is the forcing term on the right-hand side, and Y the function to be integrated, an explicit method finds $Y(t + \Delta t)$.

Explicit time integration schemes are subject to the Courant–Friedrichs–Lewy (CFL) condition for stability [CFL28, LL03]. The CFL condition requires the time step to be proportional to the smallest spatial particle resolution, which in SPH corresponds to the smallest smoothing length among all particles. In the SPH method, the CFL condition assumes the form [Mon89, Mon92]

$$\Delta t = \min_a \left\{ \frac{h_a}{c_a} \right\},$$

where c_a is the speed of sound for particle a .

5.4.1 Adaptive time step

As it can be seen from the above relationship, when the smoothing length gets smaller, the time step can become so small to be prohibitive, hence the need of a variable or *adaptive time step* [MK99, GRD⁺10]. Different formulations for the adaptive time step have been proposed over the years. The most used ones are listed in the following.

The formulation of [Mon89, Mon92] involves the CFL condition, the artificial viscosity, and the force terms:

$$\Delta t = \min \{0.25\Delta t_f, 0.4\Delta t_{cv}\},$$

or, alternatively,

$$\Delta t = 0.3 \min \{\Delta t_f, \Delta t_{cv}\},$$

where Δt_f is the part of the time step due to external forces, and Δt_{cv} is due to Courant and viscous forces. The part of the time step due to external forces is [GRD⁺10, p. 25]

$$\Delta t_f = \min_a \left\{ \sqrt{\frac{h_a}{\|\mathbf{f}_a\|}} \right\},$$

where $\|\mathbf{f}_a\|$ is the magnitude of the acceleration vector for particle a . The part of the time step due to Courant and viscous forces is

$$\Delta t_{cv} = \min_a \left\{ \frac{h}{c_s + \max_b \left| \frac{h\mathbf{v}^{ba} \cdot \mathbf{x}^{ba}}{\|\mathbf{x}^{ba}\|^2} \right|} \right\}.$$

An alternative formulation of Δt_{cv} is

$$\Delta t_{cv} = \min_a \left\{ \frac{h_a}{c_a + 0.6(\alpha_\pi c_a + \beta_\pi \max_k \{\phi_{ab}\})} \right\},$$

where α_π , β_π , and ϕ_{ab} are the terms related to artificial viscosity (see section 5), and k is the number of interacting pairs.

6 Boundary treatment

SPH has been unable to treat generalized boundary conditions, but this is not an inherent limitation of the method.

— Randles and Libersky, 1996

Since the birth of SPH, several boundary treatments have been proposed. In the early computational fluid dynamics uses of SPH, simple boundary conditions such as non-penetrating surfaces were adopted. Nonetheless, being a particle method, the boundary of

the simulation domain Ω is never well defined. The problem is that close to the boundaries of the simulation domain, the SPH method is affected by particle deficiency, since the integral of the kernel function is truncated by the boundary. This situation is illustrated in Figure 3.

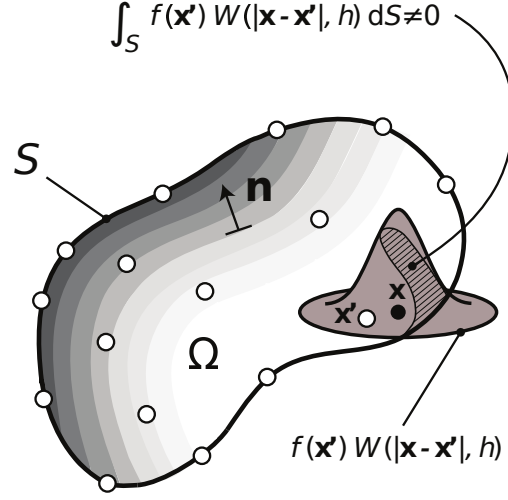


Figure 3: Behavior of SPH near the boundaries.

In order to be stable, the SPH method needs a sufficient and necessary number of particles within the support domain of κh , where κ is a coefficient to be specified. In one, two, and three dimensions the number of neighboring particles (including the particle itself) should be about 5, 21, and 57, respectively, if the particles are arranged in an initial lattice with a smoothing length of 1.2 times the particle spacing Δx , and $\kappa = 2$ [LL03]. Figure 4 illustrates this concept for the two-dimensional case.

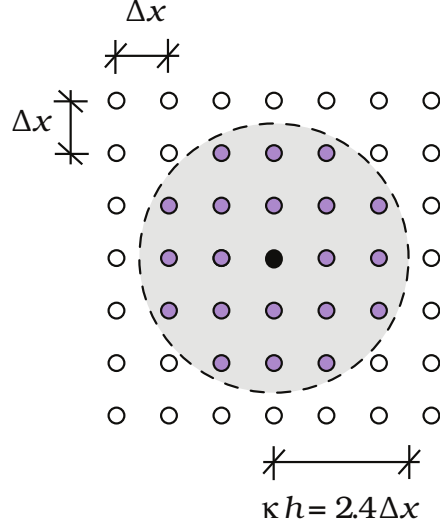


Figure 4: Stability of two-dimensional SPH.

Therefore, for particles near or on the boundary, only particles inside the domain contribute to the summation, and this one-sided contribution leads to wrong solutions.

Over the years, several authors have proposed different solutions to address the problem of boundary treatment in SPH. In all of these approaches, boundaries are typically defined by lines of particles that exert repulsive forces on fluid particles. In this section, we explore the following concepts and formulations:

- Ghost particles [LPC⁺93];
- Lennard–Jones potential for repulsive boundary forces [Mon94];
- Normal boundary forces [MK99, MKI03];
- Radial boundary forces [MK09].

6.1 Ghost particles

One of the first approaches for boundary treatment is the introduction of *ghost particles*. In [LPC⁺93], Libersky was the first to introduce ghost particles to reflect a symmetrical surface, and in [Mon94] Monaghan used a line of virtual particles located right on the solid boundary, to produce a *highly repulsive force* to the particles near the boundary, in order to avoid penetration.

[LL03] merged and improved these two approaches to treat the solid boundary conditions. According to [LL03], virtual particles (VP) can be of two types:

- Type I: right on the solid boundary (similar to [Mon94]);

- Type II: they fill in the region outside the boundary and close to it (similar to [LPC+93]).

Virtual particles of type I take part in kernel and particle approximation for the real particles, but their position and physical variables do not evolve in the simulation. They are used to exert a repulsive boundary force to prevent the interior particles from penetrating the solid boundary.

The ghost particles (i.e., VP of type II) can be applied to both *solid boundary* and *free surfaces*. They are constructed as follows. If a real particle a is located within the distance κh from the boundary, then a VP is symmetrically placed on the outside of the boundary (see Figure 5). These particles have the *same density* and *pressure* as the corresponding real particles, but *opposite velocity*. VP of type II do not evolve their parameters, since they are created symmetrically to the corresponding real particles at every time step.

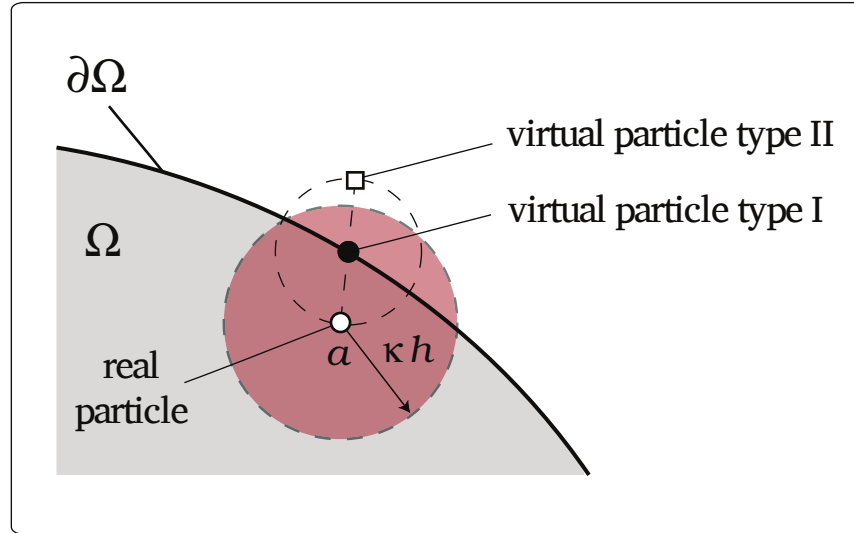


Figure 5: Construction of VP of type II according to [LL03].

Summarizing, for a particle a close to the boundary, the set of particles falling into its domain of influence can be divided into three subsets (see Figure 6):

- $I(a)$: interior or real particles;
- $B(a)$: boundary particles, or virtual particles of type I;
- $E(a)$: exterior particles, or virtual particles of type II (ghost particles).

The total number of particles N in the support domain of a particle a close to the boundary is therefore $N(a) = I(a) \cup B(a) \cup E(a)$.

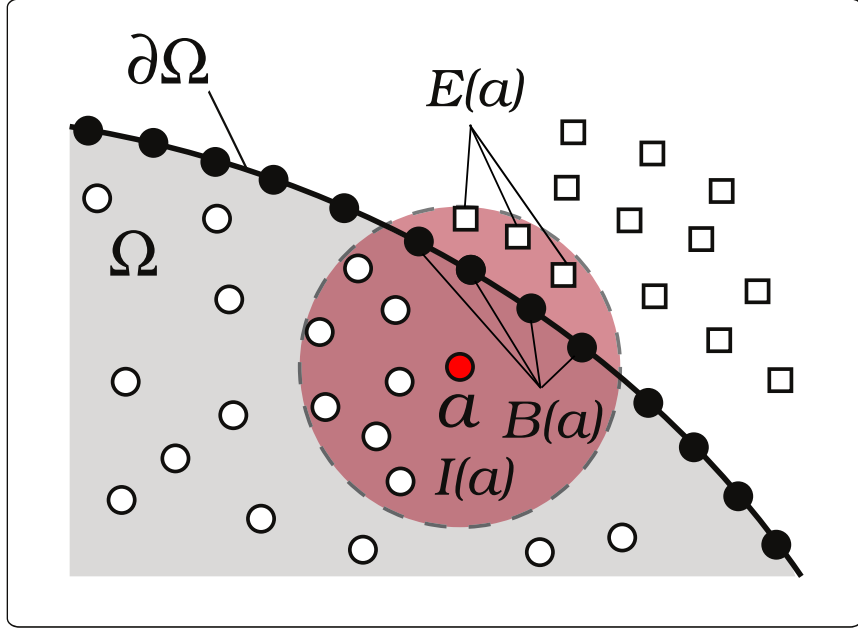


Figure 6: Types of particles falling into the support domain of a particle a close to the boundary.

6.1.1 Pros and contras

This approach has the advantages to restore the SPH consistency near the boundaries and to prevent non-physical penetration through the solid boundary. Yet in the original presentation of [RL96], a very important point was missing: *how to generate* the ghost particles. Typically, the ghost particles are generated by mirroring the fluid particles, and this implies that they must adapt with the fluid particles at each time step, causing an additional computational effort. The ghost particles become particularly unwieldy in presence of corners or surfaces with high curvature, since in these situations they cannot be placed without ambiguity.

6.2 Lennard–Jones potential

The formulation of this repulsive force is based on the known forces between molecules [Mon94]. In fact, it takes the form of a Lennard–Jones potential.

An example of Lennard–Jones potential is shown in Figure 7.

The way this potential works is quite simple. If a real particle a is approaching a boundary particle k , then a pairwise repulsive force is applied along the centerline of these

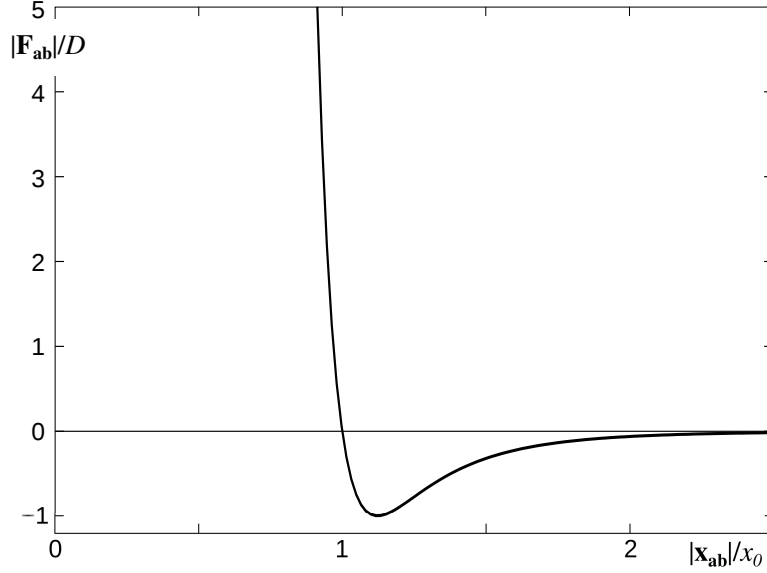


Figure 7: A graph of strength versus distance for the 12-6 Lennard-Jones potential.

two particles. The mathematical formulation is

$$\mathbf{f}_{ak} = \begin{cases} D \left[\left(\frac{x_0}{\|\mathbf{x}^{ak}\|} \right)^{n_1} - \left(\frac{x_0}{\|\mathbf{x}^{ak}\|} \right)^{n_2} \right] \frac{\mathbf{x}^{ak}}{\|\mathbf{x}^{ak}\|^2}, & \|\mathbf{x}^{ak}\| \leq x_0, \\ 0, & \|\mathbf{x}^{ak}\| > x_0, \end{cases}$$

where $\|\mathbf{x}^{ak}\|$ is the distance between particle a and boundary particle k , x_0 is a cutoff distance, D is a problem parameter, and usually $n_1 = 12$ and $n_2 = 6$. The coefficient D should be chosen considering the physical configuration. For problems involving dams, bores, weirs with fluid depth H , we may set $D = 5gH$, but also $D = 10gH$ or $D = gH$.

The cutoff distance x_0 is usually selected to be approximately equal to the initial particle spacing Δx . If it is too large, then some particles may feel the repulsive force in the initial distribution already. If it is too small, then the particles will penetrate the boundary before feeling the repulsive force.

6.2.1 Pros and contras

If the parameters are well tuned, the Lennard-Jones potential formulation avoids particle penetration into the boundary, ensuring that the velocity component normal to the boundaries vanishes. Nonetheless, Lennard-Jones forces are not satisfactory, since a particle moving parallel to the boundary is subject to a non-uniform normal force and a non-zero tangential force, leading to large disturbances in the flow near a boundary. See, for example, the shear-driven cavity problem in section 7.2. To avoid this shortcoming, some

authors adopted boundary particle forces based on an interpolation procedure that we are going to present in the next section.

6.3 Boundary force approach

According to [MKI03], “the force per unit mass \mathbf{f}_k on the boundary particles is due to the fluid particles unless the moving rigid body strikes a fixed boundary”. Neglecting the latter case for simplicity, we can write

$$\mathbf{f}_k = \sum_a \mathbf{f}_{ka},$$

where \mathbf{f}_{ka} represents the force per unit mass on boundary particle k due to fluid particles a . The force \mathbf{f}_k on each boundary particle is computed by summing up the contributions from all the surrounding water particles which fall within the supporting kernel.

In this section, we will review two different formulations available in literature for the calculation of the boundary force \mathbf{f}_{ka} : the normal force and the radial force approaches.

6.3.1 Normal force

[MKI03] proposed that the forces from neighboring boundary particles should give rise to a force normal to the boundary. Let’s consider the k -th boundary particle and the fluid particle a , as shown in Figure 8.

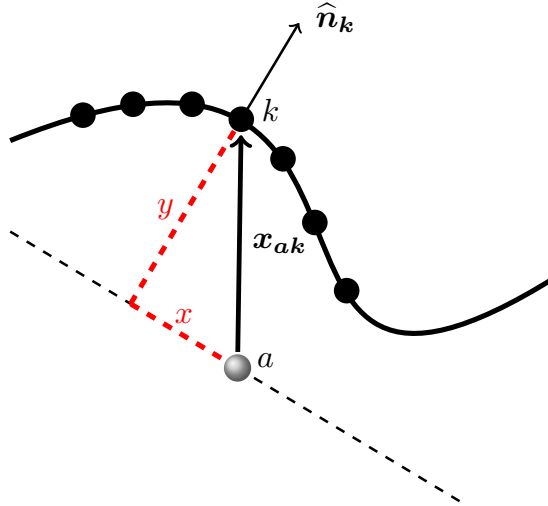


Figure 8: Normal boundary force approach.

Then the repulsive force per unit mass on boundary particle k due to fluid particle a is given by

$$\mathbf{f}_{ka} = -\frac{m_a}{m_a + m_k} B(x, y) \hat{\mathbf{n}}_k,$$

and, in turn, because of Newton's third law, the force per unit mass on a due to k is

$$\mathbf{f}_{ak} = \frac{m_k}{m_a + m_k} B(x, y) \hat{\mathbf{n}}_k. \quad (26)$$

Let us have a closer look to the parameters involved in these equations:

- $\hat{\mathbf{n}}_k$ is the outward unit normal to the boundary at the location of particle k ;
- m_a and m_k are the masses of particles a and k , respectively;
- x and y are the tangential and the normal distance between a and k , respectively;
- $B(x, y)$ is a function of the local coordinates x and y .

The normal distance y can be viewed as the projection on the direction of $\hat{\mathbf{n}}_k$ of the vector distance $\mathbf{x}^{ak} = \mathbf{x}^k - \mathbf{x}^a$ between a and k , thus it can be obtained from the scalar product

$$y = \mathbf{x}^{ak} \cdot \hat{\mathbf{n}}_k.$$

The total force per unit mass on fluid particle a due to all the boundary particles k in the support domain $B(a)$ is

$$\mathbf{f}_a = \sum_{k \in B(a)} \mathbf{f}_{ak}. \quad (27)$$

This force is then added to the SPH approximation of the equations of motion (23) with the first form of symmetrization

$$\frac{dv_i^a}{dt} = - \sum_{b=1}^N m_b \left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_i^a} + 2 \sum_{b=1}^N m_b \left(\frac{\mu_a D_{ij}^a}{\rho_a^2} + \frac{\mu_b D_{ij}^b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_j^a} + f_i^a, \quad (28)$$

with $i = 1, \dots, d$.

6.3.2 Choice of the function $B(x, y)$

We still have to define the form of the function $B(x, y)$ appearing in (26). In general, the variation of $B(x, y)$ with x should ensure that the force on a fluid particle moving parallel to the boundary is constant. Moreover, $B(x, y)$ should be chosen so that it rapidly increases as y tends to zero, i.e., as the fluid particle is approaching the boundary, to prevent penetration.

Choice 1. [MKI03] wrote $B(x, y)$ as a separable variable function

$$B(x, y) = \Gamma(y) \chi(x),$$

where

$$\chi(x) = \begin{cases} \left(1 - \frac{x}{\Delta p}\right) & \text{if } 0 < x < \Delta p, \\ 0 & \text{otherwise,} \end{cases}$$

with Δp being the spacing between the boundary particles². The function $\chi(x)$ ensures that a fluid particle moving parallel to the boundary will experience a constant force. The function $\Gamma(y)$ has a form related to the gradient of the kernel [MKI03, Mon05], i.e.,

$$\Gamma(y) = \begin{cases} \frac{2}{3} \beta & \text{if } 0 < q < \frac{2}{3}, \\ \beta \left(2q - \frac{3}{2} q^2\right) & \text{if } \frac{2}{3} < q < 1, \\ \frac{1}{2} \beta (2 - q)^2 & \text{if } 1 < q < 2, \\ 0 & \text{otherwise,} \end{cases}$$

where

- $q = y/h$, with y being the normal distance (see Figure 8);
- $\beta = 0.02 c_s^2/y$, being c_s the speed of sound. This term represents an estimate of the maximum force per unit mass necessary to stop a particle moving at the estimated maximum speed. The factor $1/y$ ensures that a faster moving particle can be stopped.

Overall, the function $\Gamma(y)$ quickly increases as y decreases in order to prevent the fluid particle from penetrating the boundary of the simulation domain.

Choice 2. A similar approach was given by [MK99], who defined \mathbf{f}_{ak} as

$$\mathbf{f}_{ak} = R(y) P(x) \hat{\mathbf{n}}_k.$$

The specific form of $R(y)$ is not crucial, however, as before, it should rapidly increase as y tends to zero. The authors proposed the following formulation:

$$R(y) = \begin{cases} A \frac{1}{\sqrt{q}} (1 - q) & \text{if } q < 1, \\ 0 & \text{otherwise,} \end{cases}$$

where

²Typically the spacing between the boundary particles Δp is about one half of the initial fluid particle spacing Δx [MK09].

- $q = y/2h$;
- A is a coefficient having dimensions of an acceleration:

$$A = \frac{1}{h} 0.01 c_a^2.$$

As before, the function $P(x)$ is designed so that a fluid particle moving parallel to the boundary will experience a constant repulsive boundary force

$$P(x) = \begin{cases} \frac{1}{2} \left(1 + \cos \frac{\pi x}{\Delta p} \right) & \text{if } x < \Delta p, \\ 0 & \text{otherwise.} \end{cases}$$

Choice 3. [GRD⁺10] introduced a correction factor $\epsilon(z, u_\perp)$ into the original formulation of [MK99]

$$\mathbf{f}_{ak} = R(y) P(x) \epsilon(z, u_\perp) \hat{\mathbf{n}}_k.$$

The function $\epsilon(z, u_\perp)$ is used to adjust the magnitude of the repulsive force according to the local water depth and velocity of the fluid particle normal to the boundary. Please note that this is the formulation that has been implemented in the SPHysics code, according to the guide of [GRD⁺10].

6.3.3 Pros and contras

The main advantage of using a normal boundary force approach over the Lennard–Jones repulsive forces is that fluid particles moving parallel to the boundary experience a constant repulsive boundary force. This, as we have seen, is one flaw of the Lennard–Jones repulsive forces which is here avoided. Unfortunately, the problem with all the normal boundary force approaches lies in the calculation of the outward normals to a surface. In fact, this is not an easy task to be achieved in SPH. Sometimes the calculation of the outward unit normal may be ambiguous, as illustrated in Figure 9.

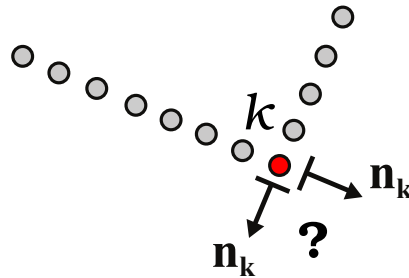


Figure 9: Ambiguity in the calculation of the outward normal at a sharp corner.

In this example, a corner particle has two outward normals, so it may interact with the fluid particles using either one or another of the normals to the lines that make up the corner, or we may assign a normal with direction halfway between the normals to the two straight lines. In any case, it is clear that, because of this ambiguity in defining the outward normal, we have to treat the corner particles differently from the other boundary particles, hence we need to allow the code to identify corner particles. This is not desirable since it inevitably increases the computational effort of the simulation.

6.3.4 Radial force

In order to solve the problem of treating the corner particles differently from the other boundary particles, [MK09] suggested to adopt repulsive boundary forces with radial direction. See Figure 10 for an illustration.

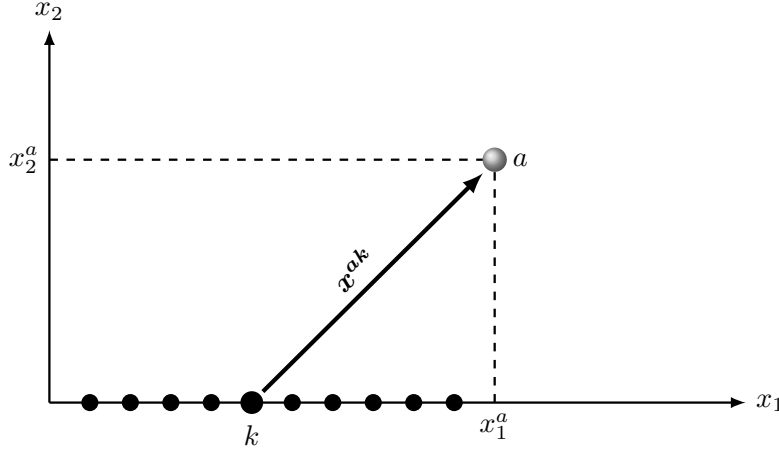


Figure 10: Radial boundary force approach.

The proposed formulation for \mathbf{f}_{ak} is

$$\mathbf{f}_{ak} = \frac{K}{\beta} \frac{\mathbf{x}^{ak}}{\|\mathbf{x}^{ak}\|^2} W(\|\mathbf{x}^{ak}\|/h) \frac{2m_k}{m_a + m_k}, \quad (29)$$

where

- $K = gD$, with D being the initial water depth;
- $\beta = \Delta x / \Delta p$. This parameter ensures that if we change the spacing between the boundary particles, the force on the fluid does not change.

If we assume that all particle masses are equal, the above equation (29) reduces to

$$\mathbf{f}_{ak} = \frac{K}{\beta} \frac{\mathbf{x}^{ak}}{\|\mathbf{x}^{ak}\|^2} W(\|\mathbf{x}^{ak}\|/h). \quad (30)$$

We then proceed as in section 6.3.1 using (27) and (28).

This approach removes the need of computing the outward unit normals to the boundaries, which as we have seen was the origin of some ambiguities in the normal boundary force formulation. Moreover, it simplifies SPH algorithms and turns out to be superior to other formulations when dealing with complicated boundaries.

7 Simulations

In this section, we present the numerical results for two classical benchmark problems: the one-dimensional shock tube [LL03, p. 156] and the two-dimensional shear-driven cavity problem [LL03, p. 94, Example 3.6].

We conducted our experiments on a laptop Lenovo ThinkPad T460s with Ubuntu 22.04.1 LTS and MATLAB R2022a installed, with Intel Core i7-6600 CPU, 20GB RAM, and Mesa Intel HD Graphics 520.

7.1 Sod shock tube problem

The Sod shock tube [Sod78, §3] is a long straight tube filled with gas, separated by a membrane into two parts of different pressures and densities. The sudden removal of the membrane generates a shock wave, a rarefaction wave, and a contact discontinuity. The shock wave moves into the region with lower density, the rarefaction wave moves into the region with higher density, and the contact discontinuity forms near the center and moves into the low-density region following the shock wave.

This is a good benchmark because it has an exact solution available, and it has also been implemented in SPHM. The script `Driver_Sod_shocktube` can be used to generate the Sod shock tube profiles using Sod’s original data.

In the simulation of this section, the initial conditions have been chosen as in [MG83, HK89], and are as follows:

$$x \in [-0.6, 0.0], \quad \rho = 1, \quad \mathbf{v} = 0, \quad e = 2.5, \quad p = 1, \quad \Delta x = 0.001875.$$

$$x \in [0.0, 0.6], \quad \rho = 0.25, \quad \mathbf{v} = 0, \quad e = 1.795, \quad p = 0.1795, \quad \Delta x = 0.0075.$$

We use 400 particles in the simulation, all having the same mass $m = 0.001875$. The shock tube length is 1.2 meters, corresponding to the interval $[-0.6, 0.6]$ on the real axis. In the left subinterval $[-0.6, 0.0]$ we place 320 evenly distributed particles, while in the right subinterval $[0.0, 0.6]$ we place 80 evenly distributed particles. The reason for this distribution is to have a discontinuous density profile along the tube. We adopt the quartic smoothing function from [LL03, p. 92]. Moreover, we use the equation of state (19) for ideal gas, with $\gamma = 1.4$. We employ a fixed time step $\Delta t = 0.005$ s, and the simulation runs for a total 40 time steps, so that the final time of the simulation is $t = 0.20$ s.

The Monaghan type artificial viscosity introduced in [Mon92] and presented in section 5 is used to resolve the shock front.

To perform this simulation and reproduce the results presented in this section, simply run the script `Driver_SPHM` with the parameter `sph.example = 1`. This will generate the mfiles in the results folder. Then, run the script `Driver_shocktube_profiles` to plot the profiles in Figure 11.

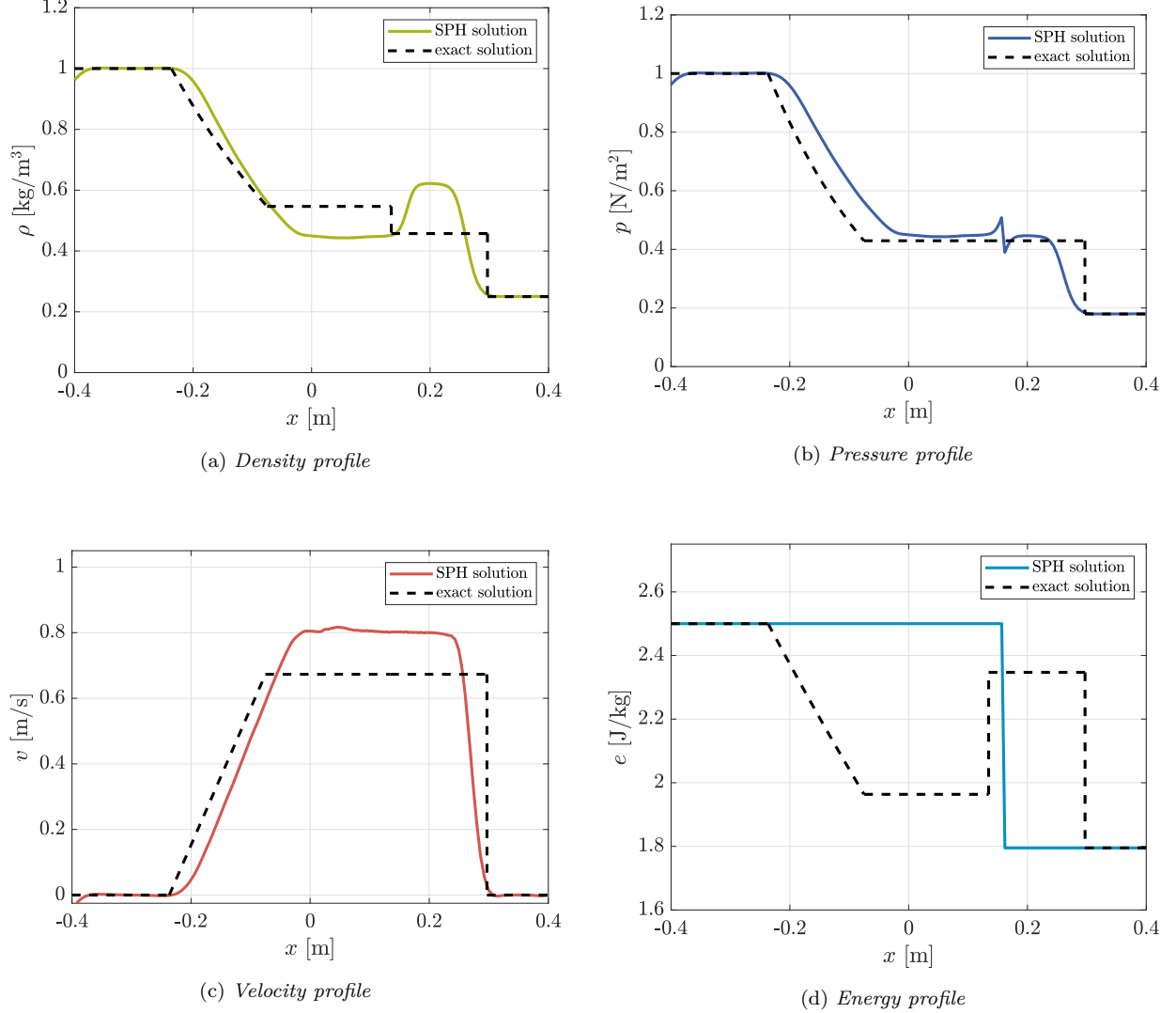


Figure 11: Profiles for the shock tube example presented in section 7.1.

The shock front is located around $x = 0.3$. The rarefaction wave is between $x = -0.3$ and $x = 0$. The contact discontinuity is located between $x = 0.1$ and $x = 0.2$.

7.2 Shear-driven cavity problem

Here, we consider the two-dimensional shear-driven cavity problem. This is the flow of a fluid in a squared section tube where the top side moves at a constant velocity v_{top} , while the other three sides are fixed. After a certain amount of time, the flow reaches a steady state, with the formation of a recirculation pattern.

For the numerical simulation in this section, the length of the side of the square domain is $\ell = 10^{-3}$ m, the viscosity is $\nu = 10^{-6}$ m²/s, and the density is $\rho = 10^3$ kg/m³. The top side of the square moves at a velocity of $v_{\text{top}} = 10^{-3}$ m/s. The initial particle distribution is shown in Figure 12. A total of 1600 real particles (the filled circles ●) are evenly distributed in the square domain, while 320 virtual particles (the gray squares ■) are used on the boundaries.

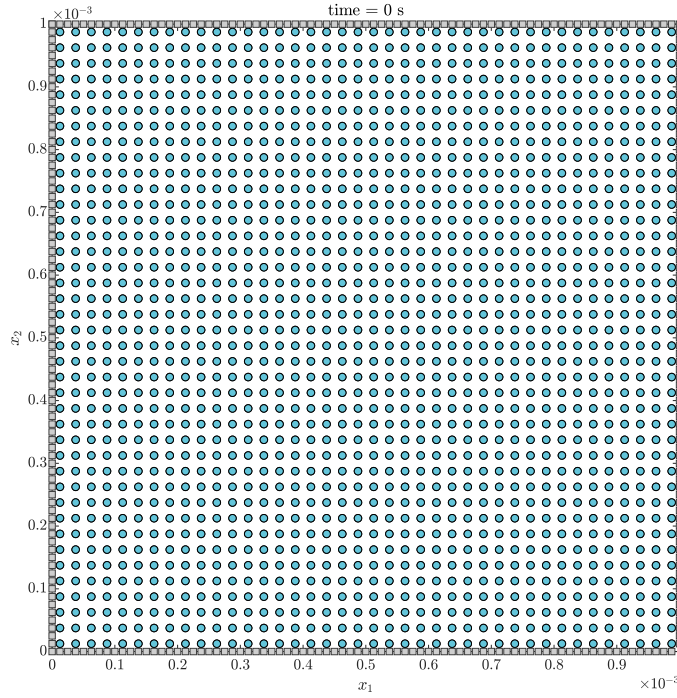


Figure 12: Initial particle distribution for the shear-driven cavity problem presented in section 7.2.

For the time evolution, we employ a constant time step $\Delta t = 5 \times 10^{-5}$, and run the simulation for a total of 10 000 time steps. Figure 13 shows the particle and the velocity distribution when the steady state has been reached. The recirculation pattern can be clearly observed in panel (b) of Figure 13. The results of this section can be reproduced by running the script `Driver_SPHM` with the parameter `sph.example = 2`. Then, to generate Figure 13, run the script `Driver_shear_cavity_steady_state` with `plt.velocity` set to 0 or 1 to reproduce panel (a) or (b), respectively.

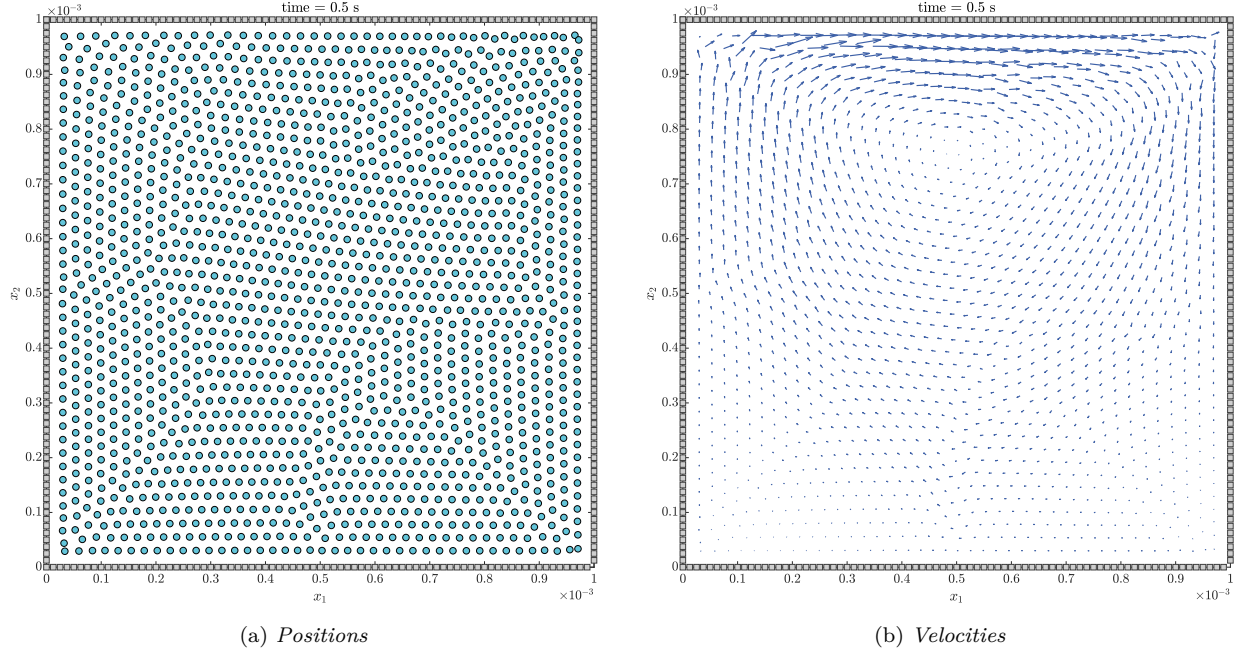


Figure 13: Steady state particle and velocity distribution for the shear-driven cavity problem presented in section 7.2.

8 Floating objects

The motion of a rigid body interacting with a fluid is determined by specifying the motion of the center of mass of the object and the rotation about the center of mass [Mon05, GRD⁺10]. This is illustrated in Figure 14, where \mathbf{X} and \mathbf{V} indicate the position and the velocity of the center of mass, respectively.

The floating object is represented by a rectangular rigid body, which is discretized by a set of boundary particles that are equally spaced around its boundary [MKI03].

8.1 Equations of motion

The equations of motion for a floating object should describe the time evolution of the center of mass, the angular velocity, and the position of the particles on the boundary of the rigid body.

Center of mass. Let B_{rb} denote the set of the boundary particles of the rigid body. The center of mass of the rigid body is evolved according to the vector equation

$$M \frac{d\mathbf{v}}{dt} = \sum_{k \in B_{\text{rb}}} m_k \mathbf{f}_k, \quad (31)$$

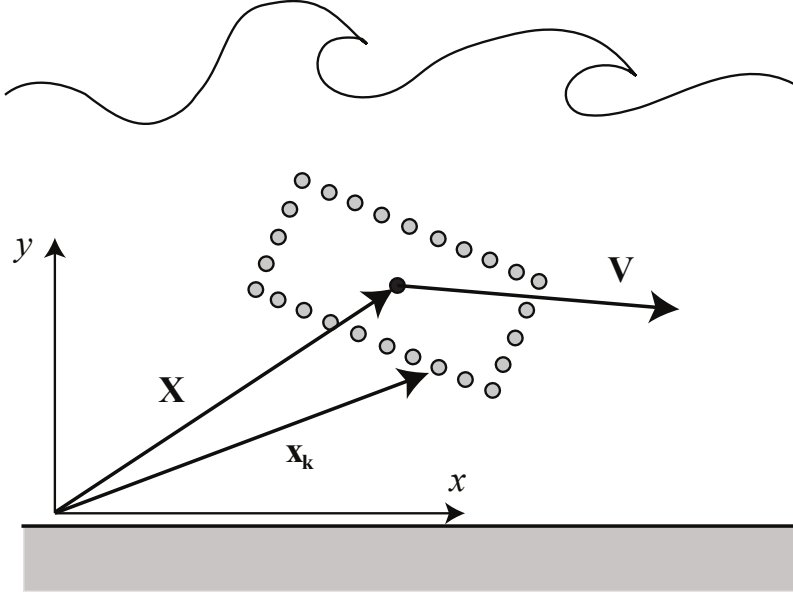


Figure 14: Floating object in a two-dimensional flow.

where M is the mass of the rigid body, m_k is the mass of the boundary particles of the rigid body and \mathbf{f}_k are the forces acting on the boundary particles of the rigid body due to the surrounding fluid. For the formulation of \mathbf{f}_k , see section 6.3.

Figures 15 and 16 sketch the concept of boundary forces calculation for a floating object by using the normal boundary force approach. Note that (31) takes care of the three translational degrees of freedom of the rigid object.

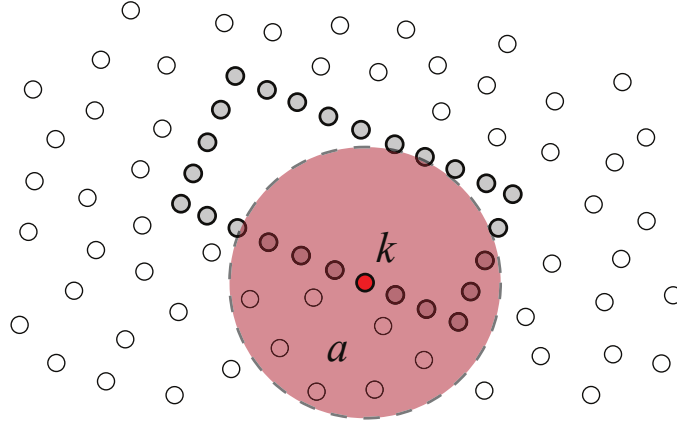


Figure 15: Calculation of the force on each boundary particle of the rigid body.

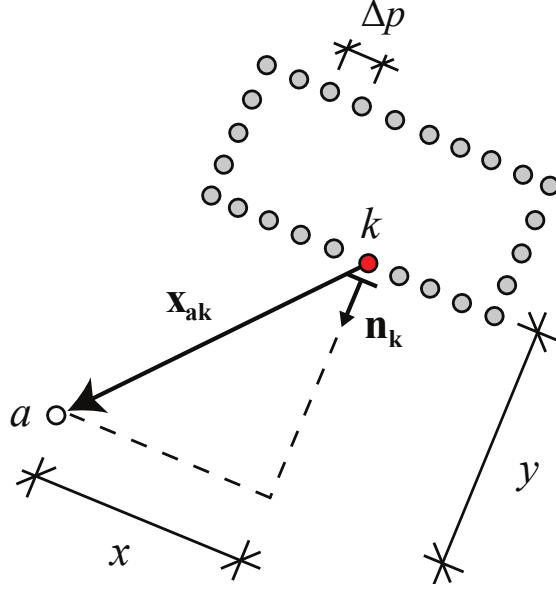


Figure 16: Calculation of the boundary force: force normal to the boundary.

Angular velocity. In the case of a two-dimensional motion, the equation for the evolution of the angular velocity $\boldsymbol{\Omega}$ is

$$I \frac{d\boldsymbol{\Omega}}{dt} = \sum_{k \in B_{rb}} m_k (\mathbf{x}^k - \mathbf{X}) \times \mathbf{f}_k, \quad (32)$$

where I is the moment of inertia and \mathbf{x}^k denotes the position of the k -th boundary particle. This equation takes care of the rotational degree of freedom of the rigid object. The term $I\boldsymbol{\Omega}$ represents the *angular momentum* (also called moment of momentum).

Boundary particles. The boundary particles are moved according to

$$\frac{d\mathbf{x}^k}{dt} = \mathbf{v} + \boldsymbol{\Omega} \times (\mathbf{x}^k - \mathbf{X}). \quad (33)$$

Equations (31), (32) and (33) are integrated in time to predict the values of \mathbf{v} , $\boldsymbol{\Omega}$ and the position \mathbf{x}^k of the boundary particles for the next time step.

The SPH approximation of the equations of motion in presence of floating rigid objects becomes

$$\frac{dv_i^a}{dt} = - \sum_{b=1}^N m_b \left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_i^a} + 2 \sum_{b=1}^N m_b \left(\frac{\mu_a D_{ij}^a}{\rho_a^2} + \frac{\mu_b D_{ij}^b}{\rho_b^2} \right) \frac{\partial W_{ab}}{\partial x_j^a} + f_i^a,$$

which is exactly equation (28) that we get in section 6.3.1 when discussing *fixed* boundaries. So the boundary force formulations that we discussed in the previous section are well suited for this kind of application.

References

- [Ben90] Benz, W. *Smooth Particle Hydrodynamics: A Review*, pages 269–288. Springer Netherlands, Dordrecht, 1990.
- [cC10] Çengel, Y. A. and Cimbala, J. M. *Fluid Mechanics: Fundamentals and Applications - second edition in SI units*. McGraw-Hill, 2010.
- [CFL28] Courant, R., Friedrichs, K., and Lewy, H. On the partial difference equations of mathematical physics. Technical report, AEC Computing and Applied Mathematics Center, 1928.
- [Col48] Cole, R. H. *Underwater Explosions*. Princeton University Press, 1948.
- [Ful94] Fulk, D. A. *A Numerical Analysis of Smoothed Particle Hydrodynamics*. PhD thesis, Air Force Institute of Technology, Sep. 1994.
- [GM77] Gingold, R. A. and Monaghan, J. J. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. Royal Astronomical Society*, 181:375–389, 1977.
- [GM82] Gingold, R. A. and Monaghan, J. J. Kernel estimates as a basis for general particle method in hydrodynamics. *J. Comput. Phys.*, 46:429–453, 1982.
- [GRD⁺10] Gesteira, M., Rogers, B., Dalrymple, R., Crespo, A., and Narayanaswamy, M. *User Guide for the SPHysics code*. 2010.
- [HK89] Hernquist, L. and Katz, N. TreeSPH: A Unification of SPH with the Hierarchical Tree Method. *The Astrophysical Journal Supplement Series*, 70:419–446, June 1989.
- [LL03] Liu, G. R. and Liu, M. B. *Smoothed Particle Hydrodynamics: A Meshfree Particle Method*. World Scientific, Singapore, 2003.
- [LPC⁺93] Libersky, L. D., Petschek, A. G., Carney, T. C., Hipp, J. R., and Allahdadi, F. A. High strain Lagrangian hydrodynamics: A three-dimensional SPH code for dynamic material response. *J. Comput. Phys.*, 109, 11 1993.
- [Mal77] Malvern, L. E. *Introduction to the Mechanics of a Continuous Medium*. Prentice Hall, 1977.
- [MG83] Monaghan, J. J. and Gingold, R. A. Shock simulation by the particle method SPH. *J. Comput. Phys.*, 52(2):374–389, 1983.
- [MK99] Monaghan, J. J. and Kos, A. Solitary Waves on a Cretan Beach. *J. Waterw. Port Coast. Ocean Eng.*, 125(3):145–154, 1999.

- [MK09] Monaghan, J. J. and Kajtar, J. B. SPH particle boundary forces for arbitrary boundaries. *Comput. Phys. Commun.*, 180:1811–1820, 2009.
- [MKI03] Monaghan, J. J., Kos, A., and Issa, N. Fluid motion generated by impact. *J. Waterw. Port Coast. Ocean Eng.*, 129:250–259, 2003.
- [Mon89] Monaghan, J. J. On the problem of penetration in particle methods. *J. Comput. Phys.*, 82(1):1–15, 1989.
- [Mon92] Monaghan, J. J. Smoothed particle hydrodynamics. *Annu. Rev. Astron. Astrophys.*, 30:543–574, 1992.
- [Mon94] Monaghan, J. J. Simulating free surface flows with SPH. *J. Comput. Phys.*, 110(2):399–406, 1994.
- [Mon05] Monaghan, J. J. Smoothed particle hydrodynamics. *Rep. Prog. Phys.*, 68:1703–1759, 2005.
- [Mon12] Monaghan, J. J. Smoothed Particle Hydrodynamics and Its Diverse Applications. *Annu. Rev. Fluid Mech.*, 44(1):323–346, 2012.
- [Mus75] Muskhelishvili, N. I. *Some Basic Problems of the Mathematical Theory of Elasticity*. Springer, 1975.
- [Noh87] Noh, W. F. Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux. *J. Comput. Phys.*, 72(1):78–120, 1987.
- [RL96] Randles, P. W. and Libersky, L. D. Smoothed particle hydrodynamics: Some recent improvements and applications. *Comput. Methods Appl. Mech. Eng.*, 139(1):375–408, December 1996.
- [RLM06] Romano, A., Lancellotta, R., and Marasco, A. *Continuum Mechanics using Mathematica - Fundamentals, Applications, and Scientific Computing*. Birkhäuser, 2006.
- [Sod78] Sod, G. A. A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws. *J. Comput. Phys.*, 27(1):1–31, 1978.