

# An Improved Dynamic Boundary Condition in SPH Method

Xu LI\*, Haiwen ZHANG\*\*, Dekui YUAN\*\*\*

\*Department of Mechanics, Tianjin University, Tianjin 300350, China

\*\*China Institute of Water Resources and Hydropower Research, Beijing 100038, China

\*\*\*Department of Mechanics, Tianjin University, Tianjin 300350, China, E-mail: dkyuan@tju.edu.cn (Corresponding author)

**crossref** <http://dx.doi.org/10.5755/j02.mech.28674>

## 1. Introduction

Smoothed particle hydrodynamics (SPH) was initially proposed by Monaghan and Gingold [1], and by Lucy [2] in 1977. SPH has progressed tremendously due to intensive theoretical work and computational improvements, particularly since the 1990s. This numerical method has been shown to be robust and applicable to a wide variety of fields. The main advantages of SPH come from its pure Lagrangian nature. The method can efficiently deal with the existence of large voids without special treatment. Also, the method can handle large deformations and extreme fragmentation without additional complicated or rather unphysical approaches. From a numerical point of view, its Lagrangian nature brings simplicity and parallelism to its algorithm. In addition, multiphase media can be easily described if the relationships between the phases are determined.

In fluid mechanics, fluids are described as discrete particles, and the governing equations are integrated into the Lagrangian form for these particles. The physical quantities of each particle can be obtained by interpolating the neighboring particles' relevant values through a kernel function. In some way, the role of kernel function is like a weight function used to obtain a weighted average of the values of the particles in the support domain. Computationally, the interpolation enables discrete information to be smoothed and continuous in the domain.

SPH methods have been successfully used in astrophysical applications and hydrodynamic problems, such as free surface flow [3, 4], gravity current [5]. SPH method can also be applied to many other scientific and engineering applications, such as diesel engines, hydraulic engineering, and nanofluid flow. Boundary condition is one of the key issues to solve these problems which is also very important for other particle methods such as MPS, and considerable efforts have been devoted to it. Different types of boundary conditions can be used, such as ghost particles [6], repulsive particles [3], LUST [7], and so on. Among these boundary conditions, dynamic boundary condition (DBC) was first developed by Crespo, Gómez and Dalrymple [8] and has been widely used. In DBC, boundary particles share the same properties with fluid particles, and they follow the same equations of state and continuity when computing the density variation and the acceleration of fluid particles. The differences are that the boundary particles are not allowed to move, or they can only move according to some external input and the density of boundary particles are kept constant. In this way, all the variables associated with the particles can be calculated in the same manner with a considerable saving of computation time.

Although DBC has been proved feasible in many

cases, we found it would bring some error due to the asymmetry of the repulsive force field. When the fluid particles collide with the boundary, they may not be reflected specularly. Although this error may be tiny, it should be avoided because the interaction between the fluid and the boundary should be controllable and precise since how the error occurs and propagates is uncertain. In addition, in some simulations, we found a few fluid particles broke through the boundary. This phenomenon is unphysical, and it possibly indicated that the repulsive force was not strong enough.

We noticed that local uniform stencil boundary condition (LUST) is presented to solve the two problems. The repulsive force and the symmetry have been improved to some extent in LUST. However, for some exceptional cases, such as waterdrop impact, these problems still exist.

This paper presents an improved dynamic boundary condition (IDBC) in SPH methods with three improvements. Firstly, the boundary interaction is enhanced to decrease the number of boundary particles and prevent fluid particles from breaking through the boundary. Secondly, the shape of the support domain attached to the boundary is modified to keep the force field close to the boundary being symmetrical. Finally, a correction factor is introduced to guarantee the unity. Meanwhile, the original kernel is retained, and the algorithm keeps concise.

## 2. Method description

The main features of the SPH method have been amply described in the references [9-11]. The fundamental idea is to consider that a function  $f(\mathbf{r})$  can be approximated by:

$$f(\mathbf{r}) = \int f(\mathbf{r}')W(\mathbf{r}-\mathbf{r}',h)d\mathbf{r}', \quad (1)$$

where:  $\mathbf{r}$  is position;  $W$  is kernel function and  $h$  is smoothing length that defines the size of the compact support. This approximation, in discrete notation, leads to:

$$f_a = \sum_b m_b \frac{f_b}{\rho_b} W_{ab}, \quad (2)$$

where:  $a$  and  $b$  are particles;  $W_{ab}$  is the kernel function between particle  $a$  and  $b$ ;  $m_k$  and  $\rho_k$  are the mass and density of particle  $k$  respectively. A cubic spline kernel developed by Monaghan and Latanzio [12] is used in this study, as follows:

$$W_{ab} = \alpha_D \begin{cases} 1 - 1.5q^2 + 0.75q^3 & \text{if } 0 \leq q \leq 1 \\ 0.25(2-q)^3 & \text{if } 1 \leq q \leq 2, \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where:  $q = r/h$  with  $r$  being the distance between particles;  $\alpha_D$  equals to  $10/7\pi h^2$  in two-dimensional (2D) conditions and  $1/\pi h^3$  in three-dimensional (3D). Due to the particular choice of the cubic spline kernel, with the first derivative versus  $q$  being zero, the tensile correction method proposed by Monaghan [13] is used to prevent particle clumping. This kernel and tensile correction have been thoroughly proved to be feasible.

The momentum conservation equation in Lagrangian form is:

$$\frac{d\mathbf{v}}{dt} = -\frac{\nabla P}{\rho} + \mathbf{g} + \mathbf{\Gamma} \quad (4)$$

where:  $\mathbf{v}$  is velocity;  $P$  is pressure;  $\rho$  is density;  $\mathbf{g}$  is gravitational acceleration and  $\mathbf{\Gamma}$  refers to dissipative terms. SPH methods describe the momentum equation with different approaches through different formulations of the dissipative terms. The artificial viscosity scheme, proposed by Monaghan [10], is a conventional method within fluid simulation using SPH due primarily to its simplicity. In SPH notation, Eq. (5) can be written as:

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left( \frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla_a W_{ab} + \mathbf{g}, \quad (5)$$

where the artificial viscosity term  $\Pi_{ab}$  is given by:

$$\Pi_{ab} = \begin{cases} \frac{-\alpha \bar{c}_{ab} \mu_{ab} + \beta \mu_{ab}^2}{\bar{\rho}_{ab}} & \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} < 0 \\ 0 & \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} > 0 \end{cases}, \quad (6)$$

with  $\mu_{ab} = \frac{h \mathbf{v}_{ab} \cdot \mathbf{r}_{ab}}{\mathbf{r}_{ab}^2 + \eta^2}$ , where  $\mathbf{r}_{ab} = \mathbf{r}_a - \mathbf{r}_b$  and  $\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b$

with  $\mathbf{r}_k$  and  $\mathbf{v}_k$  being the particle position and velocity, respectively;  $\bar{\rho}_{ab} = (\rho_a + \rho_b)/2$  and  $\bar{c}_{ab} = (c_a + c_b)/2$  are average density and average sound speed of particle a and b;  $\eta = 0.1h$ ;  $\alpha$  and  $\beta$  are constants.  $\alpha$  is related to shear and bulk viscosity.  $\beta$  handles high Mach number shocks and is roughly equivalent to the Von Neumann-Richtmyer viscosity used in finite-difference methods. Following Monaghan [10],  $\beta$  is considered to be zero.

The continuity equation can be replaced either by the interpolant:

$$\rho_a = \sum_b m_b W_{ab}, \quad (7)$$

or by:

$$\frac{d\rho_a}{dt} = \sum_b m_b \mathbf{v}_{ab} \cdot \nabla_a W_{ab}. \quad (8)$$

Eq. (7) comes from Eq. (2) in SPH interpolation. Eq. (8) is the continuity equation in SPH form, in which the density change is computed by solving the conservation of mass [9]. In this study, Eq. (8) is used to calculate the fluid density in order to avoid the artificial density decrease near fluid interfaces.

Following Monaghan [5], Batchelor [14] and Cole [15], pressure can be calculated by the following expression:

$$P = B \left[ (\rho/\rho_0)^\gamma - 1 \right], \quad (9)$$

where the constant of state  $\gamma = 7$ ,  $B = c_0^2 \rho_0 / \gamma$ ,  $\rho_0 = 998.1 \text{ kg/m}^3$  (at the temperature of 20 °C) is the reference density and  $c_0 = c(\rho_0) = \sqrt{\partial P / \partial \rho}|_{\rho_0}$  is the speed of sound at the reference density. Particles are moved using the XSPH method according to Monaghan [10, 16]:

$$\frac{d\mathbf{r}_a}{dt} = \mathbf{v}_a + \varepsilon \sum_b \frac{m_b}{\bar{\rho}_{ab}} \mathbf{v}_{ab} W_{ab}, \quad (10)$$

where:  $\varepsilon = 0.5$ . With this method, the particle is moved depending on not only its own velocity but also the average velocity of the neighboring particles. The XSPH variant has proven useful in the simulation of nearly incompressible fluids such as water, where it keeps the particles orderly in the absence of viscosity.

The time step is controlled by the Courant condition, the force terms, and the viscous diffusion term [16]. A variable time step  $\Delta t$  is calculated according to Monaghan and Kos [4]:

$$\Delta t_f = \min_a \left( \frac{h}{\sqrt{|\mathbf{f}_a|}} \right) \quad \Delta t_{cv} = \min_a \frac{h}{c_s + \max_b |\mu_{ab}|}, \quad (11)$$

$$\Delta t = CFL \cdot \min(\Delta t_f, \Delta t_{cv})$$

in which:  $\Delta t_f$  is based on the force per unit mass  $\mathbf{f}$ , actually it is equivalent to the magnitude of particle acceleration  $\mathbf{a}_a$  and  $\Delta t_{cv}$  combines the Courant and the viscous time-step controls.

For numerical simulations, the boundary condition is one of the key issues. In SPH method, boundary conditions need to be treated specially. When a particle approaches the boundary, the number of neighbors decreases because there is nothing on the other side of the boundary. The lack of neighbors could result in unrealistic effects as some variables are highly dependent on the summation of the variables attached to the neighbors. To solve this problem, many boundary conditions are developed [3, 4, 7, 17]. One of the successful boundary conditions is dynamic boundary condition [8]. In this method, all the particles, no matter which types they are, follow the same equations of continuity and state, but the positions of the boundary particles do not change unless external forces are imposed. One attractive advantage of this assumption is that the implication is simplified, and computational time can be considerably saved. This boundary condition, first presented by Dalrymple and Knio [18], has been tested [8] and widely used [19, 20].

### 3. Improved dynamic boundary condition

Dynamic boundary condition has been fully described by Crespo, Gómez, and Dalrymple [8]. The basic idea is to calculate the pressure of the boundary particles with the equation of state. The equation of state can be obtained from the first term of the Taylor expansion of Eq. (9):

$$P_a = c^2 (\rho_a - \rho_{a0}), \quad (12)$$

where:  $c$  is the speed of sound, which is assumed as a constant and  $\rho_{a0}$  is the initial density of particle  $a$ .

To simplify the description of the problem, only two particles are considered here, i.e., one boundary particle and one fluid particle. In the absence of viscosity and gravity, the momentum equation for the fluid particle (Particle A) becomes:

$$\frac{dv_a}{dt} = -m_b \left( \frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} \right) \nabla_a W_{ab}. \quad (13)$$

Combining Eq. (13) with Eq. (12) gives:

$$\frac{dv_a}{dt} = -m_b c^2 \left( \frac{\rho_b - \rho_0}{\rho_b^2} + \frac{\rho_a - \rho_0}{\rho_a^2} \right) \nabla_a W_{ab}. \quad (14)$$

When there are only two particles, Eq. (7) can be written as:

$$\rho_a = m_a W_{aa} + m_b W_{ab}; \quad \rho_b = m_a W_{ba} + m_b W_{bb}. \quad (15)$$

As we know,

$$W_0 = W_{aa} = W_{bb} = W(r=0). \quad (16)$$

Assuming all the particles have the same mass  $m$ , we have:

$$\rho_a = \rho_b = \rho = m(W_{ab} + W_0); \quad \rho_0 = mW_0. \quad (17)$$

Thus Eq. (14) becomes:

$$\frac{dv_a}{dt} = -2c^2 \frac{W_{ab}}{(W_{ab} + W_0)^2} \nabla_a W_{ab}. \quad (18)$$

With the artificial viscosity as described in Eq. (6), tensile correction, gravity and more particles, we obtain:

$$\frac{dv_a}{dt} = -\sum_b \left( 2c^2 \frac{W_{ab}}{(W_{ab} + W_0)^2} + m_b \Pi_{ab} + f_{ten} \right) \nabla_a W_{ab} + \mathbf{g}, \quad (19)$$

where the tensile correction term  $f_{ten}$  is defined according to Monaghan [13]:

$$f_{ten} = (R_a + R_b) \left[ \frac{W_{ab}}{W} \left( \frac{2}{3} \right) \right]^4 \quad (20)$$

$$R_k = \frac{\varepsilon P_k}{\rho_k} \quad \varepsilon = \begin{cases} 0.01 & \text{if } P_k > 0 \\ -0.2 & \text{else} \end{cases}$$

#### 3.1. Improvements on particle interaction

Although the dynamic boundary condition has been widely used and tested, there are still some problems. In some situations, some particles can break through the boundary in an unphysical way. A possible method to solve this problem is to enhance the forces between fluid particles and boundary particles. In Eq. (12), the pressure is expressed using the first-order term of the Taylor expansion of the equation of state. Since the exponent  $\gamma = 7$  in Eq. (9), the equation of state can be expanded into seventh order, as follows:

$$P_a = c_0^2 \rho_0 \lambda \left( 1 + 3\lambda + 5\lambda^2 + 5\lambda^3 + 3\lambda^4 + \lambda^5 + \frac{1}{7}\lambda^6 \right), \quad (21)$$

where:  $\lambda = (\rho_a - \rho_0) / \rho_0$ .

In another way, Eq. (21) can be considered as a binomial expansion if we express it in the following form:

$$P_a = \frac{c_0^2 \rho_0}{\gamma} \left[ \left( \frac{\rho_a - \rho_0}{\rho_0} + 1 \right)^\gamma - 1 \right]. \quad (22)$$

With the same assumption, Eq. (19) can be expressed as:

$$\frac{dv_a}{dt} = -\sum_b \left[ 2c^2 \frac{W_{ab}}{(W_{ab} + W_0)^2} f_\theta + m_b \Pi_{ab} + f_{ten} \right] \nabla_a W_{ab} + \mathbf{g};$$

$$f_\theta = 1 + 3\theta + 5\theta^2 + 5\theta^3 + 3\theta^4 + \theta^5 + \frac{1}{7}\theta^6;$$

$$\theta = \frac{W_{ab}}{W_0}. \quad (23)$$

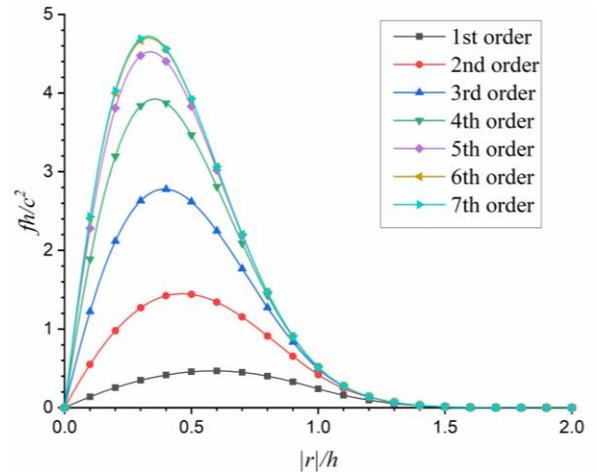


Fig. 1 Dimensionless force between one boundary particle (Particle a) and one fluid particle (Particle b) in different orders

Considering the particular case of a cubic kernel, the differences of the repulsive forces among different orders of the equation are shown in Fig. 1.

As shown in Fig. 1, the higher-order expansion does enhance the repulsive forces, but the difference between the sixth order and the seventh order is not significant. One reason is that the coefficient of the seventh order is one-seventh in Eq. (23). Another reason is that the base  $\theta$  is less than 1 and the exponent increases which make the repulsive force decreased dramatically in the seventh order. Another interesting phenomenon is that with the higher-order, the curve's peak becomes higher and moves closer to the boundary particle. It means that an energetic particle needs more energy to climb over the peak and this particle must get close enough to the boundary. Therefore, the repulsive force of the boundary can be enhanced to prevent fluid particles

from breaking through the solid boundary, meanwhile it exerts little impact on all the fluid particles. Only the particles close enough to boundary particles will be repulsed significantly. The total potential energy created by the repulsive force can be represented by integrating the dimensionless force over the dimensionless distance from 0 to 2 as shown in Fig. 1. The dimensionless potential energy in different orders is shown in Fig. 2.

With the full expansion, the potential energy is increased by more than seven times. It is an important index to evaluate how IDBC works. Using the full expanding form makes all the fluid and boundary particles follow the same equation of state. However, considering the computation efficiency, we need to determine how many orders should be kept according to each order's strength.

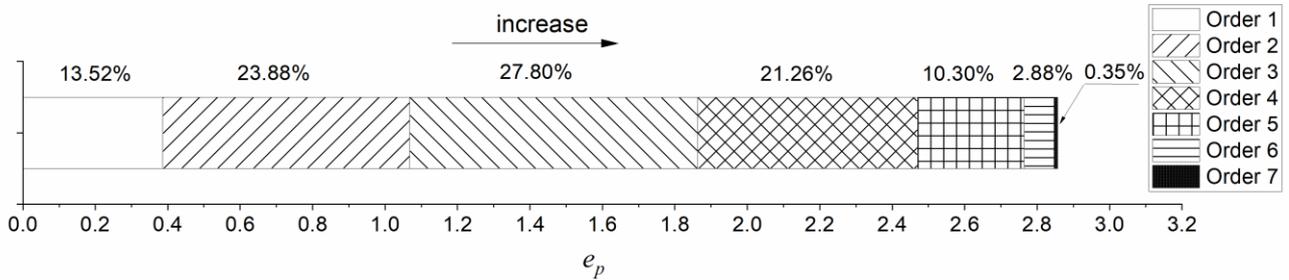


Fig. 2 The distribution of the dimensionless potential energy in each order

### 3.2. Improvement on support domain

Another problem is that the field of repulsive forces is not uniform and it is also not specular. According to DBC, the boundary described as a group of particles cannot move, but the forces are also discretized. The field of repulsive force is circular because the support domain of the particles is a circle. The superposition of the force fields of the boundary particles results in the edge of the whole force field being unsmoothed. When fluid particles reach the boundary, they may not be reflected specularly. For some cases, such as droplet impact, the result will be asymmetrical [21]. For an ideal boundary, it is unphysical and unacceptable. Furthermore, an uneven force field may lead to some errors out of control.

We set a given fluid particle at  $(x_0, y_0)$  and some neighbor boundary particles distribute from  $(x_0 - \Delta x, y_B)$  to  $(x_0 + \Delta x, y_B)$ . The density  $\rho_B$  of each boundary particle is the same, and the pressure  $P_B$  is calculated using the equation of state. We can express the repulsive force on the fluid particle as Eq. (24):

$$\begin{aligned} \mathbf{f} &= f_x \mathbf{i} + f_y \mathbf{j} \\ r &= |\mathbf{r}| = \sqrt{(x_0 - x)^2 + (y_0 - y_B)^2} \\ f_x &= \int_{x_0 - \Delta x}^{x_0 + \Delta x} \frac{P_B}{\rho_B} \frac{x_0 - x}{r} \frac{\partial W(r)}{\partial r} dx \\ f_y &= \int_{x_0 - \Delta x}^{x_0 + \Delta x} \frac{P_B}{\rho_B} \frac{y_0 - y_B}{r} \frac{\partial W(r)}{\partial r} dx \end{aligned} \quad (24)$$

From Eq. (24), it is evident that  $f_x \equiv 0$  and  $f_y$  is only dependent on the distance in the normal direction ( $y_0 - y_B$ ).  $\mathbf{f}$  can be expressed as:

$$\mathbf{f} = \left( \int_{x_0 - \Delta x}^{x_0 + \Delta x} \frac{P_B}{\rho_B} \frac{y_0 - y_B}{r} \frac{\partial W(r)}{\partial r} dx \right) \mathbf{j}. \quad (25)$$

When Eq. (25) is expressed in SPH form, the summation of the repulsive forces could not be independent of  $x$  if the distribution of the particles is not symmetrical. Thus, we introduced IDBC to make it only related to the distance in the normal direction.

Therefore, we change the form of the support domain of the boundary particles in IDBC. In theory, a boundary particle represents a group of corresponding boundary elements. The contour line of the repulsive force should be parallel to the boundary so that the shape of the support domain is changed from a circle to a rectangle, as shown in Fig. 3.

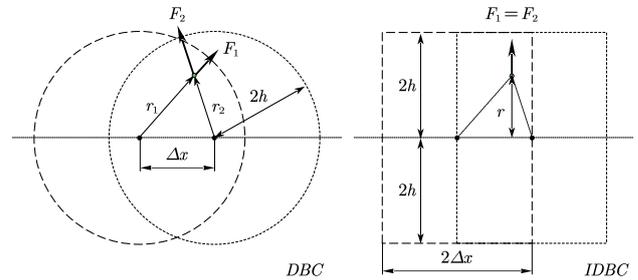


Fig. 3 Two different ways to calculate the distance between the fluid particles and the boundary particles. The forces  $F$  depend on this distance

In the compilation, when calculating the distances between fluid particles and boundary particles, the projection in the normal direction of the boundary is only considered. Then the fluid particles at the same distance to the

boundary are repulsed equally. The length in the normal direction is  $4h$ , and the length in the tangential direction is two times the distance between the particles, as shown in Fig. 3.

Fluid particles will search neighbor boundary particles in the rectangle support domain. The distance  $r_1$  and the gradient of kernel function  $(\nabla_a W_{ab})_1$  are calculated with Eq. (26):

$$r_1 = r_{ab} \cdot \frac{\mathbf{n}}{|\mathbf{n}|} \quad (\nabla_a W_{ab})_1 = \frac{\mathbf{n}}{|\mathbf{n}|} \frac{\partial W(r_1)}{\partial r_1}, \quad (26)$$

where:  $\mathbf{n}$  is the normal vector at the boundary. They are used in the governing equations between fluid particles and boundary particles. Eq. (26) is applied to determine the density variation and compute the repulsive force for fluid particles.

As is known, the coefficient of the kernel function is related to the support domain. This coefficient  $\alpha_D$  is used to keep the integration of the kernel function  $W(r, h)$  within the support domain  $S$  equal to 1 as expressed in Eq. (27):

$$\alpha_D \int_S W(r, h) dr = 1. \quad (27)$$

The coefficient should be adjusted with the change of support domain. To simplify the compilation, a correction coefficient  $\alpha_{DB}$  on  $\alpha_D$  only for boundary particles is used:

$$\alpha_{DB} \alpha_D \int_S W(r, h) dr = 1. \quad (28)$$

For the cubic kernel function shown in Eq. (3),  $\alpha_{DB}$  is  $7\pi h/30\Delta x$  in 2D and  $\pi h^2/6\Delta x\Delta y$  in 3D.

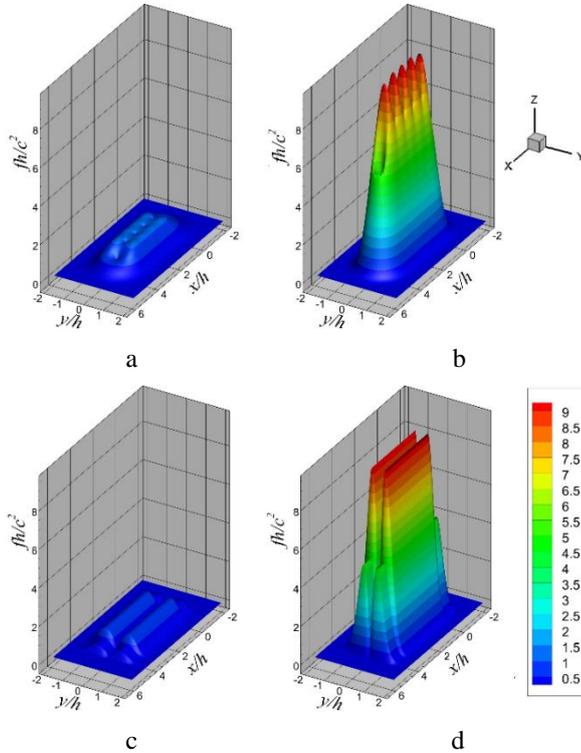


Fig. 4 The equipotential plane of the dimensionless forces field in DBC and IDBC. a) DBC; b) DBC with seventh order expansion; c) DBC with the new shape of the support domain; d) IDBC with seventh order expansion

We select six particles in a row as the boundary and calculate the repulsive force generated by them in a program which is specially developed with C++. The dimensionless forces field in DBC and IDBC with different orders are shown in Fig. 4. The expansion with higher order enhances the repulsive force obviously. This enhancement will prevent fluid particles from breaking through the boundary. The equipotential plane is smoothed by the new shape of the support domain, especially near the boundary. Furthermore, the peak of the equipotential plane is transformed into a ridge. This improvement can make the reflection specular. The form of the kernel function is kept in the vertical direction, ensuring zeroth or first-order consistency at the boundary's edge.

#### 4. Model performance test

There are some open-source codes for SPH, such as SPHysics [22] and DualSPHysics [23]. However, they are too sophisticated for our simple test cases. For convenience, we developed a program based on C++ and OpenMP to test DBC and IDBC. LUST was tested using DualSPHysics.

##### 4.1. Test case 1: Single-particle moving inside a box

The movement of a single particle inside a box is the case used to test the performances of different boundary conditions. With this case, we can make it clear that how the boundary particles repulse the single fluid particles. DBC has been proved to be useful and acceptable in an individual case, as shown in Fig. 5, in which the fluid particle hits the boundary vertically.

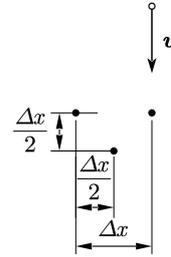


Fig. 5 Sketch of the individual case of DBC. Solid dots are the boundary particles, and the hollow one is the fluid particle

As we know, not all the particles get close to the boundary vertically or symmetrically. In DBC, the field of repulsive force is not uniform. If a particle rushes to the boundary obliquely, the field passed through by the particle is not conservative in DBC, as shown in Fig. 6 (up). It can be seen that the field passed through by the particle is asymmetric in DBC, so is the repulsive force. As a result, in DBC the fluid particle is not reflected specularly, and the energy is not conservative.

With the modified shape of the support domain and the correction factor, this problem is solved in IDBC. As shown in Fig. 6 (down), the modified field in IDBC is symmetric. A test case is designed to compare the differences between DBC and IDBC, as shown in Fig. 7. In a square box with the length of every side being 60 m, a particle moves from  $(x_0, y_0) = (30, 36)$  m with an initial velocity  $\mathbf{v} = (20, -20)$  m/s. Viscosity and gravity are neglected. The CFL number is 0.05. To make the differences distinct in a short time,

the speed of sound  $c$  close to the real speed in water is set to 1460 m/s. Although this high value of sound speed is not usually used in simulations, it is high enough to make the differences distinguishable without other unphysical assumptions as the error accumulates over time.

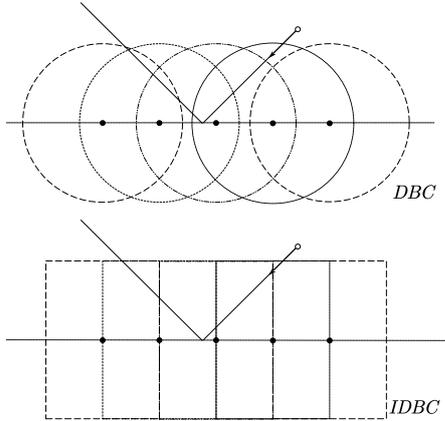


Fig. 6 The particle rushes to the boundary obliquely. The big circles and big rectangles represent the boundary particles' support domain in DBC and IDBC, respectively. Solid dots are boundary particles, and the hollow one is the fluid particle

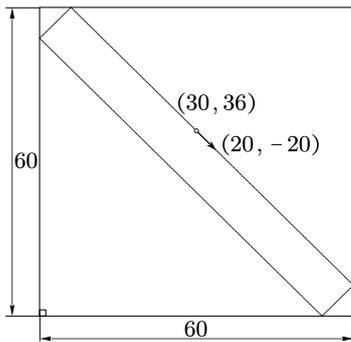


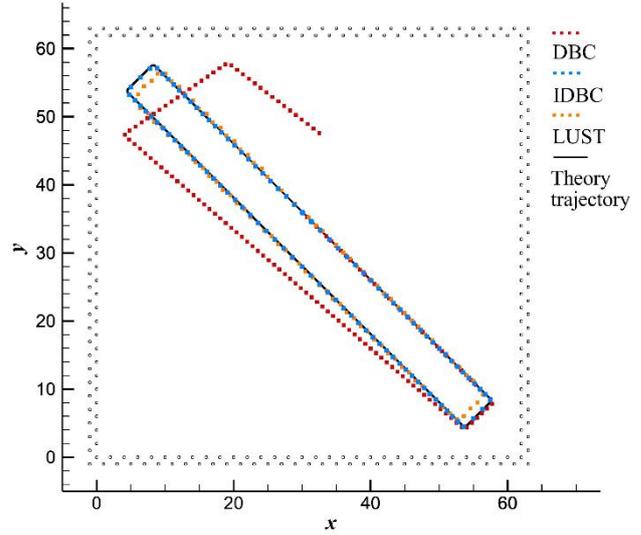
Fig. 7 Sketch of the initial setting in the test case for checking the performance on the symmetry of different boundary conditions. The dashed line is the expected route of the fluid particle

In this case, the fluid particle rushes to the boundary at 45 degrees in the square box. If the fluid particle is reflected specularly, this particle's trajectory should be a closed rectangle, as shown in Fig. 7. If the boundary generates any error, it will be amplified over time and illustrated by the fluid particle's trajectory. The error will bring unphysical fluctuation to the variables, such as kinetic energy. The results are shown in Fig. 8.

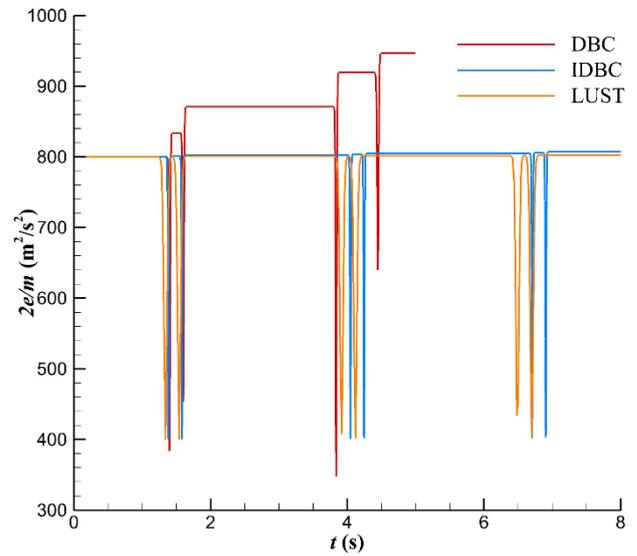
From Fig. 8, it is seen that IDBC is more accurate than DBC for this case. In this extreme test, the fluid particle cannot return the initial position after only one loop in DBC, and the energy fluctuates unphysically. However, IDBC reflects the fluid particle symmetrically and decreases the error generated by the boundary. In the LUST test, the fluid particle was also reflected symmetrically but earlier than that in DBC and IDBC. This phenomenon indicates that the repulsive force generated by LUST is stronger than DBC and IDBC when the particle is at a distance larger than  $1.0h$  to the boundary.

In general, the smoothing radius is chosen to include enough neighbors to limit the impacts of the misalignments between the particles. Therefore, the single particle

test case is useful to test the performance of the improved boundary treatment method in detail.



a



b

Fig. 8 The simulated results of the movement of a fluid particle in a box with DBC and IDBC. a) the trajectory of the fluid particle; b) the kinetic energy of the fluid particle

#### 4.2. Test case 2: Waterdrop Impact

Waterdrop is an ideal example to test IDBC with a group of fluid particles. This case describes a small waterdrop smashes to the boundary vertically driven by gravity. For this simple test case, if there is no other error, the result should be symmetrical. The waterdrop with a radius of 1.0 mm drops from the height of 2.0 mm with an initial velocity  $v = (0, -2)$  m/s, the artificial viscosity coefficient  $\alpha = 0.2$  for fluid particles and  $\alpha = 0.$  between fluid particles and boundary particles. The CFL number is 0.05. The speed of sound is 50 m/s, which is selected to guarantee that the system is weakly-compressible and the computational efficiency is acceptable simultaneously.

To distinguish the differences between DBC, LUST, and IDBC, we set the boundary particles in two ways, as shown in Fig. 9. One is symmetrical and another

one is asymmetrical. The separation between the particles,  $\Delta x$ , is 0.2 mm. The displacement between the particles of the two kinds of boundaries is  $0.3 \Delta x$ .

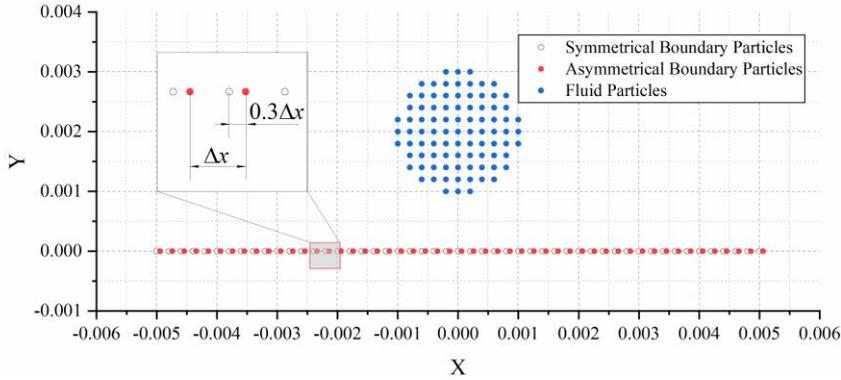


Fig. 9 Particles set in two ways: symmetrical and asymmetrical

The simulated results at  $t = 0.002$  s are shown in Fig. 10. Within one frame, the result is compared with its own mirror image to show its symmetry directly. If the boundary condition does not involve any error, the results

should not be changed by the different deployment of the boundary particles. It is shown that no matter how the particles are deployed, the results are almost the same with IDBC, but with DBC or LUST, they are not.

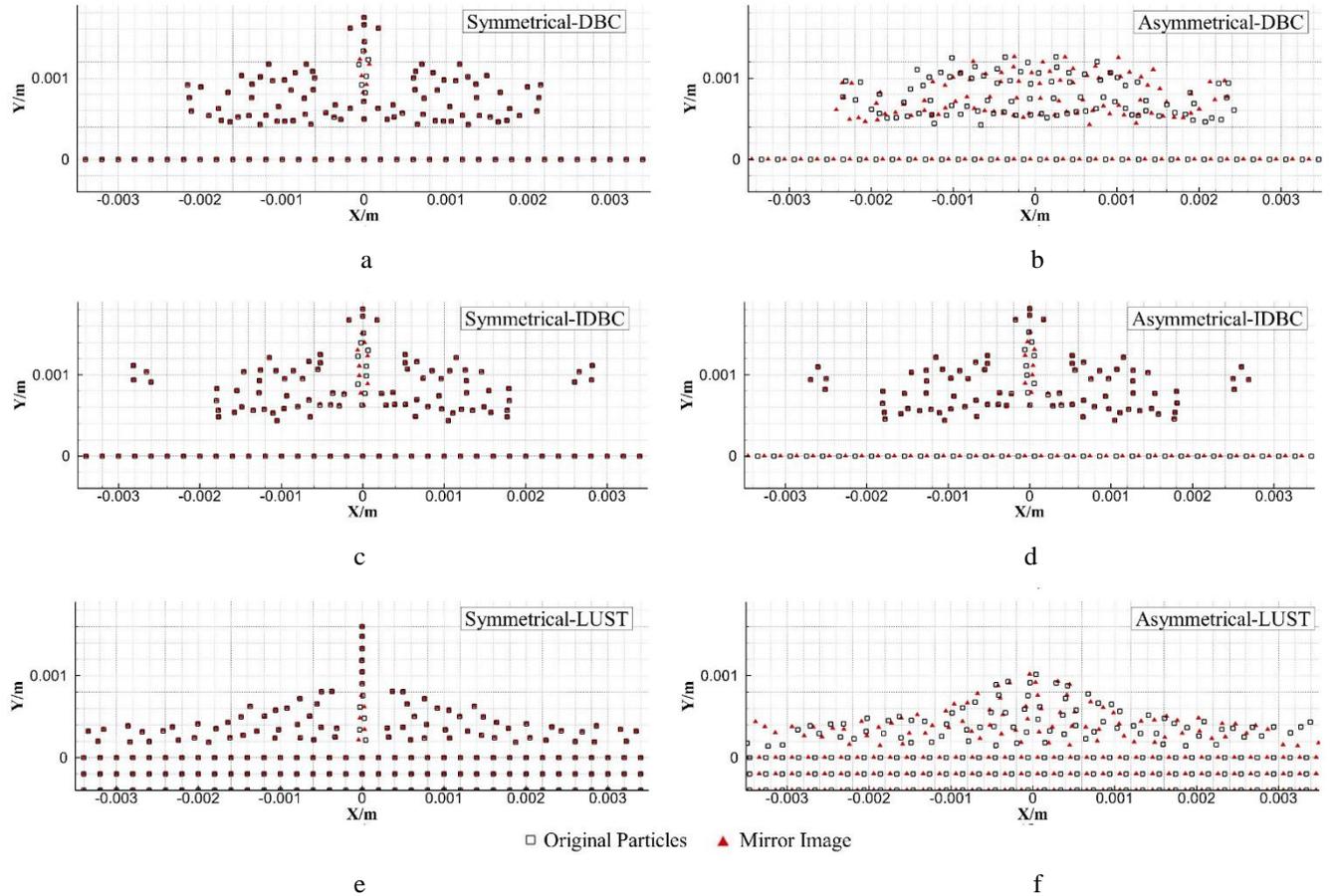


Fig. 10 The distribution of particles at  $t = 0.002$  s with different boundary conditions: a) symmetrical deployment with DBC; b) asymmetrical deployment with DBC; c) symmetrical deployment with IDBC; d) asymmetrical deployment with IDBC; (e) symmetrical deployment with LUST; (f) asymmetrical deployment with LUST

For further evaluation of the performance of DBC, LUST and IDBC, we defined a dimensionless coefficient  $f_{sym}$  as follows:

$$f_{sym} = \frac{\sqrt{f_x^2 + f_y^2}}{N\Delta x}; \quad f_x = \sum_i x_i^3; \quad f_y = \sum_i x_i y_i^2, \quad (29)$$

where:  $N$  is the number of fluid particles. This coefficient describes how symmetrical these results are and how the error generated by the boundary conditions affects the accuracy. The results are shown in Fig. 11.

As shown in Fig. 11, all three methods work well with the symmetrical deployment of boundary particles.

However, when the boundary particles are deployed asymmetrical (in terms of the waterdrop), the differences between DBC, LUST, and IDBC become obvious. In theory, no matter the boundary particles are symmetrical or not, the results should not be changed too much with the same boundary condition. In IDBC, the result with the symmetrical boundary is close to that with asymmetrical boundary. However, in DBC and LUST, obvious differences appear. These results indicate that with the asymmetrical boundary, the error generated by DBC or LUST is larger than that by IDBC.

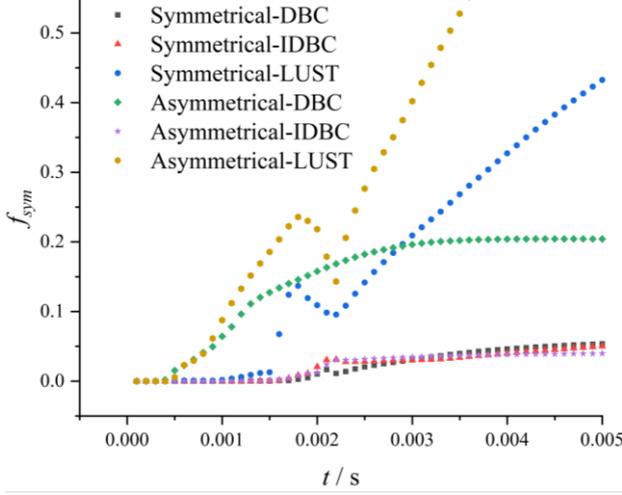


Fig. 11 Dimensionless coefficient  $f_{sym}$  of the results with DBC and IDBC

To test the strength of these three boundary conditions, we also tried to increase the velocity of the waterdrop. With LUST, some particles begin to break through the boundary at 16 m/s, and all the particles escape at 50 m/s. At 65 m/s particles begin to break through in DBC while IDBC still works well that all the fluid particles are reflected into the fluid domain symmetrically. We should notice that in LUST three layers of boundary particles are deployed, and LUST also introduced several fictitious particles for each fluid particle. Although DBC and LUST work perfectly for many other cases, the performance of IDBC is better in symmetry and repulse force.

## 5. Conclusions

Improved dynamic boundary condition (IDBC) is developed to solve two problems in DBC. One problem is that the particle may break through the boundary in some situations, such as thin boundary and high-velocity cases.

$$I_{2D} = \int_S W(r, h) dr = \int_0^{2\pi} d\theta \int_0^{2h} W(r, h) r dr \xrightarrow{q=r/h} \int_0^{2\pi} d\theta \int_0^2 W(q) q h^2 dq = \frac{7\pi h^2}{10}. \quad (c)$$

Thus,

$$\alpha_D = \frac{1}{I_{2D}} = \frac{10}{7\pi h^2}. \quad (d)$$

$$I'_{2D} = \int_S W(r, h) dr = \int_0^{2\Delta x} dx \int_0^{2h} W(r, h) dr \xrightarrow{q=r/h} \int_0^{2\Delta x} dx \int_0^2 W(q) h dq = 3h\Delta x. \quad (e)$$

Another problem is the error generated by the unphysical rough boundary. Three improvements were made in IDBC to solve these problems: a) expansion with a higher order, b) a new shape of the support domain, and c) a correction factor introduced for the kernel function. With these improvements, the potential energy produced by the repulsive force is increased over seven times, and the force field is smoothed.

The validity of the improvements has been firstly tested with a simple case that describes a single particle's movement in a box. In this case, the particle rushes to the boundary obliquely. The performance of DBC is disappointing that the particle departs the expected trajectory in a few steps. This error is decreased remarkably in IDBC.

The validity of the improvements has been further tested with a waterdrop case. This case is tested with different deployments of boundary particles and different boundary conditions. Either the deployment of the boundary particles is symmetrical or asymmetrical, IDBC keeps the results almost the same and stronger than LUST or DBC.

## Acknowledgments

This study has been supported by the Major scientific and technological projects and projects of Tianjin (18ZXRHSF00270) and the National Natural Science Foundation of China (11872271).

## Appendix: derivation of the coefficient $\alpha_{DB}$

The kernel function should follow Eq. (a):

$$\alpha_D \int_S W(r, h) dr = 1. \quad (a)$$

For cubic spline kernel:

$$W(r, h) = \begin{cases} 1 - 1.5\left(\frac{r}{h}\right)^2 + 0.75\left(\frac{r}{h}\right)^3 & \text{if } 0 \leq \frac{r}{h} \leq 1 \\ 0.25\left(2 - \frac{r}{h}\right)^3 & \text{if } 1 \leq \frac{r}{h} \leq 2. \\ 0 & \text{otherwise} \end{cases} \quad (b)$$

For the original circle support domain in 2D:

For the rectangle support domain in 2D:

and the kernel function should follow:

$$\alpha_{DB} \alpha_D \int_S W(r, h) dr = 1. \quad (f)$$

Thus,

$$I_{3D} = \int_S W(r, h) dr = \int_0^\pi d\varphi \int_0^{2\pi} d\theta \int_0^{2h} W(r, h) r^2 \sin\varphi dr \xrightarrow{q=r/h} \int_0^\pi \sin\varphi d\varphi \int_0^{2\pi} d\theta \int_0^2 W(q) q^2 h^3 dq = \pi h^3, \quad (h)$$

and

$$I'_{3D} = \int_S W(r, h) dr = \int_0^{2\Delta x} dx \int_0^{2\Delta y} dy \int_0^{2h} W(r, h) dr \xrightarrow{q=r/h} \int_0^{2\Delta x} dx \int_0^{2\Delta y} dy \int_0^2 W(q) h dq = 6h\Delta x\Delta y. \quad (i)$$

Thus,

$$\alpha_D = \frac{1}{I_{3D}} = \frac{1}{\pi h^3} \quad (j)$$

and

$$\alpha_{DB} = \frac{1}{\alpha_D I'_{3D}} = \frac{\pi h^2}{6\Delta x\Delta y}. \quad (k)$$

## References

1. **Gingold, R. A.; Monaghan, J. J.** 1977. Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Monthly Notices of the Royal Astronomical Society* 181(3): 375-89. <https://doi.org/10.1093/mnras/181.3.375>.
2. **Lucy, L. B.** 1977. A numerical approach to the testing of the fission hypothesis, *The Astronomical Journal* 82: 1013-24. <https://doi.org/10.1086/112164>.
3. **Monaghan, J. J.** 1994. Simulating free surface flows with SPH, *Journal of Computational Physics* 110(2): 399-406. <https://doi.org/10.1006/jcph.1994.1034>.
4. **Monaghan, J.; Kos, A.** 1999. Solitary waves on a Cretan beach, *Journal of Waterway, Port, Coastal, and Ocean Engineering* 125(3): 145-55. [https://doi.org/10.1061/\(asce\)0733-950x\(1999\)125:3\(145\)](https://doi.org/10.1061/(asce)0733-950x(1999)125:3(145)).
5. **Monaghan, J.; Cas, R.; Kos, A.; Hallworth, M.** 1999. Gravity currents descending a ramp in a stratified tank, *Journal of Fluid Mechanics* 379: 39-69. <https://doi.org/10.1017/s0022112098003280>.
6. **Randles, P.; Libersky, L.** 1996. Smoothed particle hydrodynamics: some recent improvements and applications, *Computer Methods in Applied Mechanics and Engineering* 139(1-4): 375-408. [https://doi.org/10.1016/s0045-7825\(96\)01090-0](https://doi.org/10.1016/s0045-7825(96)01090-0).
7. **Fourtakas, G.; Dominguez, J. M.; Vacondio, R.; Rogers, B. D.** 2019. Local Uniform Stencil (LUST) boundary condition for arbitrary 3-D boundaries in parallel smoothed particle hydrodynamics (SPH) models, *Computers & Fluids* 190: 346-361. <https://doi.org/10.1016/j.compfluid.2019.06.009>.
8. **Crespo, A.; Gómez-Gesteira, M.; Dalrymple, R. A.** 2007. Boundary conditions generated by dynamic particles in SPH methods, *CMC-TECH Science Press* 5(3): 173-184. <https://doi.org/10.3970/cmc.2007.005.173>.
9. **Monaghan, J. J.** 1982. Why particle methods work, *SIAM Journal on Scientific and Statistical Computing* 3(4): 422-33. <https://doi.org/10.1137/0903027>.
10. **Monaghan, J. J.** 1992. Smoothed particle hydrodynamics, *Annual Review of Astronomy and Astrophysics* 30(1): 543-574. <https://doi.org/10.1146/annurev.aa.30.090192.002551>.
11. **Liu G.; Liu M.** 2003. Smoothed particle hydrodynamic s: a meshfree particle method: World Scientific. <https://doi.org/10.1142/9789812564405>
12. **Monaghan, J. J.; Lattanzio, J. C.** 1985. A refined particle method for astrophysical problems, *Astronomy and Astrophysics* 149: 135-43.
13. **Monaghan, J. J.** 2000. SPH without a tensile instability, *Journal of Computational Physics* 159(2): 290-311. <https://doi.org/10.1006/jcph.2000.6439>.
14. **Batchelor, G. K.** 2000. An introduction to fluid dynamics: Cambridge University Press. <https://doi.org/10.1017/CBO9780511800955>.
15. **Cole, R. H.; Weller, R.** 1948. Underwater explosions, *Physics Today* 1(6): 35. <https://doi.org/10.1063/1.3066176>.
16. **Monaghan, J.** 1989. On the problem of penetration in particle methods, *Journal of Computational Physics* 82(1): 1-15. [https://doi.org/10.1016/0021-9991\(89\)90032-6](https://doi.org/10.1016/0021-9991(89)90032-6).
17. **Band, S.; Gissler, C.; Ihmsen, M.; Cornelis, J.; Peer, A.; Teschner, M.** 2018. Pressure boundaries for implicit incompressible SPH, *ACM Transactions on Graphics (TOG)* 37(2): 14. <https://doi.org/10.1145/3180486>.
18. **Dalrymple, R. A.; Knio, O.** 2001. SPH modelling of water waves. *Coastal Dynamics' 01*. [https://doi.org/10.1061/40566\(260\)80](https://doi.org/10.1061/40566(260)80).
19. **Crespo, A.; Gómez-Gesteira, M.; Dalrymple, R. A.** 2007. 3D SPH simulation of large waves mitigation with a dike. *Journal of Hydraulic Research* 45(5): 631-42. <https://doi.org/10.1080/00221686.2007.9521799>.
20. **Zhang, F.; Crespo, A.; Altomare, C.; Domínguez, J.;**

- Marzeddu, A.; Shang, S. P.** et al. 2018. DualSPHysics: a numerical tool to simulate real breakwaters, *Journal of Hydrodynamics* 30(1): 95-105.  
<https://doi.org/10.1007/s42241-018-0010-0>.
21. **Fang, H.; Bao, K.; Wei, J.; Zhang, H.; Wu, E.; Zheng, L.** 2009. Simulations of droplet spreading and solidification using an improved SPH model, *Numerical Heat Transfer, Part A: Applications* 55(2): 124-43.  
<https://doi.org/10.1080/10407780802603139>.
22. **Gomez-Gesteira, M.; Rogers, B. D.; Crespo, A. J.; Dalrymple, R. A.; Narayanaswamy, M.; Dominguez, J. M.** 2012. SPHysics—development of a free-surface fluid solver—Part 1: Theory and formulations. *Computers & Geosciences* 48:289-99.  
<https://doi.org/10.1016/j.cageo.2012.02.029>.
23. **Crespo, A. J.; Domínguez, J. M.; Rogers, B. D.; Gómez-Gesteira, M.; Longshaw, S.; Canelas, R.** et al. 2015. DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH), *Computer Physics Communications* 187:204-16.  
<https://doi.org/10.1016/j.cpc.2014.10.004>.

X. Li, H. Zhang, D. Yuan

## AN IMPROVED DYNAMIC BOUNDARY CONDITION IN SPH METHOD

### S u m m a r y

Dynamic boundary condition (DBC) has been widely used in SPH method. However, in certain situations, it was found that a few fluid particles could break through the boundary or were not reflected specularly. Of course, these phenomena are unphysical. To improve the performance of DBC, an improved dynamic boundary condition (IDBC) was presented in this paper. To prevent fluid particles from breaking through the boundary, the repulsive force of boundary particles was enhanced by expanding the equation of state into a higher order. To deal with the asymmetry of DBC, a rectangular support domain attached to boundary particles and a corresponding correction factor are proposed. The results of three test cases showed that the performance of IDBC was satisfied.

**Keywords:** dynamic boundary condition, DBC, SPH.

Received March 18, 2021

Accepted December 07, 2021



This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (<http://creativecommons.org/licenses/by/4.0/>).