



Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

**ScienceDirect**

Comput. Methods Appl. Mech. Engrg. 377 (2021) 113681

**Computer methods  
in applied  
mechanics and  
engineering**

[www.elsevier.com/locate/cma](http://www.elsevier.com/locate/cma)

# Fluid–rigid-body interaction simulations and validations using a coupled stabilized ISPH–DEM incorporated with the energy-tracking impulse method for multiple-body contacts

Mitsuteru Asai<sup>a,\*</sup>, Yi Li<sup>a</sup>, Bodhinanda Chandra<sup>b,\*</sup>, Shinsuke Takase<sup>c</sup>

<sup>a</sup> Department of Civil Engineering, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan

<sup>b</sup> Department of Civil and Environmental Engineering, University of California, Berkeley, CA, 94720, United States

<sup>c</sup> Department of Civil Engineering, Hachinohe Institute of Technology, 88-1 Obiraki Myo Hachinohe, Aomori 031-8501, Japan

Received 30 October 2020; received in revised form 7 January 2021; accepted 7 January 2021

Available online xxxx

## Abstract

In this paper, a new particle-based fluid–rigid-body interaction simulator for violent free-surface flow problems is developed. The incompressible Smoothed Particle Hydrodynamics (ISPH) method has been proven to produce a smooth and accurate pressure distribution of free-surface fluid flow with breaking and fragmentation. Computed hydrodynamic forces can be applied onto rigid bodies, which may simultaneously experience contact or impact with the surrounding wall boundaries or another rigid body. Modeled by using the discrete element method (DEM), the contact force between rigid bodies is traditionally calculated employing the penalty approach, where a spring-based repulsive force is approximated at the vicinity of contact points depending on the deepest penetration depth. However, for high-speed collision problems involving a system of many rigid bodies, the values of approximated repulsive forces may be highly overestimated, and thus, a much smaller time step and an excessive damping parameter are often required to stabilize the approximated forces. This problem is highly inefficient for the computational resources of the fluid–rigid body interaction simulation since the computational cost at each time step is mostly dominated by the incompressible fluid simulation. The capability to increase the time increment following the critical time step of the fluid solver is, therefore, strongly demanded to increase the simulation efficiency. The current paper incorporates the usage of the energy-tracking impulse (ETI) method as an alternative approach to handle contact accurately. To achieve better energy conservation and enhance stability, Stronge's hypothesis is considered instead of the generally assumed Newton's contact law. The current work also covers three experimental validation tests, which were conducted to assure the quality and robustness of the coupled ISPH–DEM implementation.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

**Keywords:** Smoothed Particle Hydrodynamics; Discrete Element Method; Fluid–structure interaction; Rigid-body dynamics; Contact; Validation and verification

## 1. Introduction

The Smoothed Particle Hydrodynamics (SPH) method was originally proposed by Lucy [1] and further developed by Gingold and Monaghan [2] for treating astrophysical problems. The method's main advantage is the absence of

\* Corresponding authors.

E-mail addresses: [asai@doc.kyushu-u.ac.jp](mailto:asai@doc.kyushu-u.ac.jp) (M. Asai), [bchandra@berkeley.edu](mailto:bchandra@berkeley.edu) (B. Chandra).

computational grids or meshes as it spatially discretizes the analytical domain into a finite number of Lagrangian moving particles. As one of the mesh-less methods, SPH has been utilized in various engineering applications, mostly in the analysis of moving discontinuities and large deformation systems such as the free-surface flows with breaking, splash, and fragmentation; those which are difficult to predict by using traditional mesh-based methods due to its mesh entanglement and distortion issues. An explicit SPH formulation for free surface flows was first introduced by [3] and known as the weakly compressible SPH (WCSPH). Following that, a semi-implicit formulation for incompressible flow problems, often called as the Incompressible SPH (ISPH), was proposed by introducing the projection method [4]. A similar semi-implicit particle-based simulation method named Moving Particle Semi-implicit method (MPS) was already proposed by Koshizuka [5] a few years earlier to ISPH.

Application of the particle-based methods is further enhanced to simulate flow-induced scouring [6], suspended sediment transport [7], permeable porous media flow [8–10] and many other complicated engineering phenomena, e.g. [11]. These simulations were performed mostly by using the SPH and MPS since they are recognized to be well-suited for multi-physics applications as highlighted by [12,13]. One of the major targets of the multi-physics simulation is the fluid–structure interaction (FSI) between free-surface flows with either rigid [14,15] or flexible [16,17] structures. The capability and robustness of particle-based FSI simulations are attracting attention not only for engineering applications but also for computer graphics purposes [18]. The particle-based simulation can simply detect and model the contact of several solids or rigid objects. For examples, [19] suggested a coupling scheme between SPH and the Distributed Contact Discrete Element Method (SPH–DCDEM) to simulate complex fluid-multi-rigid body interactions of dam breaks with several box arrangements, whereas [20] improved the accuracy and easy handling of rigid-body geometry in 3D by introducing the gradient and Laplacian corrected SPH.

Discrete Element Method (DEM) [21] introduces the penalty-based technique as a contact response because of their simplicity of formulation. Notable applications of the penalty-based technique are not only found in DEM but also in multi-body dynamics (MBD), discontinuous-deformation analysis (DDA) [22] and the finite-element method (FEM) [23]. This approach generally allows the simulated objects or points to penetrate any physical boundaries before applying an approximated repulsive “spring” force, which normally depends on the deepest penetration distance between two contacting or colliding surfaces. Regardless of their simple formulation, the penalty-family methods also suffer from certain numerical drawbacks, which often cause instability, particularly for problems involving heavy objects as the repulsive contact forces may be highly overestimated. As a result, they generally require a much smaller time step, and an excessive damping parameter is often set to stabilize the generated forces.

To overcome the main drawbacks of the penalty-based technique, the constraint-based method [24,25] was introduced to the computational graphics community. The constraint-based methods fundamentally solve the unilateral constraint problem by assuming a linear complementarity problem (LCP) with several types of constraint conditions. For instance, Baraff [25] imposed a relative-acceleration constraint on each contact point, whereas Stewart [26] selected a relative-velocity constraint condition. The LCP becomes a considerably good choice if the main simulation objects are in resting contact because impenetrability constraints can be sufficiently applied to prevent surface penetration. However, as soon as the contact involves friction, the constraint problem becomes a nonlinear complementarity problem (NCP), and the LCP and NCP formulations combined become generally difficult to solve, as the system matrix becomes non-symmetrical and non-positive definite.

The last major way to model contact forces is known as the impulse method [27], which mostly incorporates Newton’s impact law to represent energy dissipation during contact. Mirtich [28] extended the impulse method with the velocity constraint condition and computed the impulse using Stronge’s hypothesis [29], instead of the Newton’s impact law. This modification ensures Mirtich’s method obeys energy conservation laws during frictional collisions. In the family of impulse methods, it is no longer required to solve any linear equation, although a series of contact iterations is needed at each time step. The iteration procedure is continued until the constraint conditions are satisfied at all contact points. The impulse-based methods generally require further treatment for solving multiple contact problems, which experience concurrent contacts among multiple bodies and multiple points, such as for edge-to-surface and surface-to-surface contacts. Following that, Tang and colleagues proposed an energy-tracking impulse (ETI) method [30] for multiple rigid bodies modeled by polygons. Their proposed ETI method follows the Stronge’s hypothesis to ensure the conservation of energy similar to Mirtich’s impulse method. The ETI method contains a sub-cycling as an additional iteration, which is divided into a *compression* and an *energy-release* phase to ensure proper energy storing and releasing process during contact. Most recently, Li et al. [31] proposed a more

stable and suitable extension of the ETI method for particle-discretized rigid body simulations, where the simulations were verified with both analytical solutions and experimental data. The major advantage of the impulse methods is that, no artificial material parameters are required to model the contact between rigid objects. The methods also allow a larger time increment in comparison to the simpler penalty-based approaches, and thus, providing better efficiency for FSI simulations as highlighted earlier.

In the current work, a particle-based fluid–rigid-body interaction simulator is proposed in a partitioned two-way coupling manner. Here, the ISPH method is utilized to model the free-surface flow, and is preferred rather than the conventional WCPH as it is able to produce more reliable and accurate velocity and pressure profile without any artificial viscosity term [32]. On the other hand, the energy-conservative ETI method is employed to resolve frictional contact problems between the involved rigid bodies and boundaries more accurately and efficiently. The contact response approach in the particle-discretized rigid bodies has a similarity with the cluster DEM model [19,33,34], though the ETI approach in the proposed formulation allows a similar size of time increment as the one assumed within the fluid solver using ISPH. This correspondingly reduces the overall computational cost for the coupled simulation.

The current paper is structured as follows. First, Section 2 introduces the fundamental methodology of the ISPH method along with the governing equations of the incompressible fluid flow. Section 3 provides the details of the proposed staggered two-way fluid–rigid-body coupling scheme. Once hydrodynamics and contact forces among rigid bodies and their surrounding environments are determined, the position and kinematic variables of the rigid bodies can be updated following the Newton–Euler equations. In Section 4, some validation tests, which compare the proposed formulation with conducted experimental tests, are provided. Three experiments were designed and tested to assure the accuracy of the fluid–rigid-body coupling scheme for both single and multiple objects. A follow-up numerical example to demonstrate the robustness of the method to simulate fluid–rigid-body interaction of arbitrary complex-shaped objects is presented at the end of the section. Finally, Section 5 presents the conclusions and suggestions for future works.

## 2. Smoothed Particle Hydrodynamics (SPH) methodology

The fundamentals of SPH formulation [1,2] along with the governing equations and the Chorin’s projection scheme [35] for incompressible flow are summarized. The SPH method essentially works by discretizing a continuum as a set of Lagrangian particles which interact with each other through a radial-basis kernel function. In the current work, the stabilized Incompressible SPH (ISPH) method [36] is utilized to overcome errors of artificial pressure fluctuation and density representation. This stabilization scheme appends a relaxed density-invariance term to the divergence-free pressure Poisson equation, which will be solved implicitly through a prediction–correction scheme.

### 2.1. Governing equation

The governing equations of the fluid flow, the continuity and the Navier–Stokes equation, are represented as

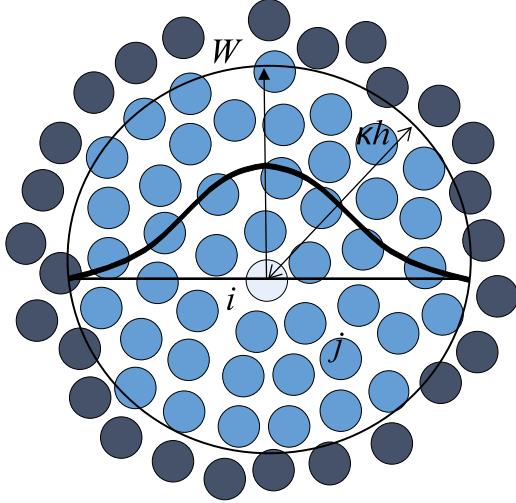
$$\frac{\partial \rho}{\partial t} + \rho \nabla \cdot \mathbf{v} = 0, \quad (1)$$

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{v} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} + \mathbf{g}, \quad (2)$$

where  $\mathbf{v}$ ,  $p$ ,  $\rho$ , and  $\nu$  denote the velocity field, pressure field, material density, and kinematic viscosity of the fluid, respectively. Whereas,  $\mathbf{g}$  and  $t$  indicate the gravitational acceleration vector and time. The turbulence stress term  $\boldsymbol{\tau}$  is also taken into consideration to represent the dissipation of energy due to turbulence in a coarse spatial discretization. In the most general incompressible flow approach, the density is assumed to be a constant value,  $\rho = \bar{\rho}$ , and thus, the aforementioned equations can be rewritten as,

$$\nabla \cdot \mathbf{v} = 0, \quad (3)$$

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\bar{\rho}} \nabla p + \nu \nabla^2 \mathbf{v} + \frac{1}{\bar{\rho}} \nabla \cdot \boldsymbol{\tau} + \mathbf{g}. \quad (4)$$



**Fig. 1.** Particle approximation in the SPH method. Here,  $i$  and  $j$  denote the particle of interest and its surrounding neighbors lying inside a support domain  $\kappa h$ .

## 2.2. Chorin's projection scheme

Eq. (4) can be solved semi-implicitly through a prediction and correction temporal discretization scheme known as the Chorin's projection method [35]. In the projection method, the velocity and pressure coupled problems are solved separately, assuming an intermediate state of velocity  $\mathbf{v}^*$ . From hereon, the superscripts  $n$  and  $n + 1$  will be used to indicate the current and future time step. The intermediate velocity  $\mathbf{v}^*$  and the prediction–correction scheme of the Navier–Stokes equation (4) are written as:

$$(\text{Predictor}) : \frac{\mathbf{v}^* - \mathbf{v}^n}{\Delta t} = \nu \nabla^2 \mathbf{v}^n + \frac{1}{\bar{\rho}} \nabla \cdot \boldsymbol{\tau}^n + \mathbf{g}^n, \quad (5)$$

$$(\text{Corrector}) : \frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} = -\frac{1}{\bar{\rho}} \nabla p^{n+1}. \quad (6)$$

The future pressure value here can be obtained by taking the divergence of the corrector step (6) and by substituting the continuity equation (3) correspondingly. The pressure Poisson equation (PPE) can then be obtained:

$$\nabla^2 p^{n+1} = \bar{\rho} \frac{\nabla \cdot \mathbf{v}^*}{\Delta t}. \quad (7)$$

From the corrected velocity  $\mathbf{v}^{n+1}$ , the future position of fluid particle can be calculated by considering an Euler integration as,

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t (\mathbf{v}^{n+1}). \quad (8)$$

## 2.3. Spatial discretization using SPH approximation

In the SPH, a physical scalar function  $\phi(\mathbf{x}_i, t)$  at a sampling point  $\mathbf{x}_i$  and time  $t$  can be represented by the following volume integration,

$$\phi(\mathbf{x}_i, t) = \int W(\|\mathbf{x}_i - \mathbf{x}_j\|, h) \phi(\mathbf{x}_j, t) dV = \int W(r_{ij}, h) \phi(\mathbf{x}_j, t) dV, \quad (9)$$

where  $W$  is a weight function called the smoothing kernel function. Within the smoothing kernel function,  $r_{ij}$  and  $h$  are the norm distance between neighbor particles and the smoothing length, respectively (see Fig. 1). Here, the distance  $r_{ij} = \|\mathbf{r}_{ij}\|$ , where  $\mathbf{r}_{ij} (= \mathbf{x}_i - \mathbf{x}_j)$  is the relative coordinate of particles  $i$  and  $j$ . Despite many possible choices of kernel functions  $W$ , the current work utilizes the cubic spline truncated within  $2h$  as proposed by [37,38].

In its discrete form, the physical scalar function  $\phi$  in (9), the gradient  $\nabla\phi$ , and the Laplacian  $\nabla^2\phi$  can be approximated as,

$$\phi(\mathbf{x}_i, t) \approx \langle \phi_i^n \rangle = \sum_j \frac{m_j}{\rho_j} \phi_j W(r_{ij}, h), \quad (10)$$

$$\nabla\phi(\mathbf{x}_i, t) \approx \langle \nabla\phi_i^n \rangle = \frac{1}{\rho_i} \sum_j m_j (\phi_j - \phi_i) \nabla W(r_{ij}, h), \quad (11)$$

$$= \rho_i \sum_j m_j \left( \frac{\phi_j}{\rho_j^2} + \frac{\phi_i}{\rho_i^2} \right) \nabla W(r_{ij}, h), \quad (12)$$

$$\nabla^2\phi(\mathbf{x}_i^n, t) \approx \langle \nabla^2\phi_i^n \rangle = \sum_j m_j \left( \frac{\rho_i + \rho_j \frac{\mathbf{r}_{ij} \cdot \nabla W(r_{ij}, h)}{r_{ij}^2 + \eta^2}}{\rho_i \rho_j} \right) (\phi_i - \phi_j). \quad (13)$$

Here, the subscripts  $i$  and  $j$  indicate the particle label, for example,  $m_j$  and  $\rho_j$  denote the representative mass and density of a neighbor particle  $j$ . Meanwhile,  $\eta$  is a parameter to avoid a zero denominator given by  $\eta^2 = 0.0001h^2$ . Note that the triangle bracket  $\langle \cdot \rangle$  here indicates the SPH approximation of a particular variable. Furthermore, notice that the two expressions of the gradient models, (11) and (12), have a manifold property that can be converted to each other analytically.

#### 2.4. Stabilized ISPH formulation

In the original ISPH, even if the PPE (7) is evaluated implicitly through a Chorin's projection method, unrealistic pressure fluctuations still cannot be eliminated. In order to overcome this problem, a stabilization term proposed by [36] is added to the PPE. One can modify Eq. (7) to,

$$\langle \nabla^2 p_i^{n+1} \rangle = \frac{\bar{\rho}}{\Delta t} \langle \nabla \cdot \mathbf{v}_i^* \rangle + \alpha \frac{\bar{\rho} - \langle \rho_i^n \rangle}{\Delta t^2}, \quad (14)$$

where the second term of the right-hand-side corresponds to the stabilization term which was derived considering particle density representation. If the particle distribution is in uniformity, the difference between the particle (or numerical) density  $\langle \rho_i^n \rangle$  and the incompressible material density  $\bar{\rho}$  is vanishingly small, and thus, removing the effect of the stabilization term. In the current work, the relaxation coefficient is chosen to be  $\alpha = 0.01$  as suggested by [36]. Note that, the stabilization term has a role of controlling the uniformity of particle's position without any additional computational effort, such as particle shifting [39].

Considering Eqs. (10)–(13), the SPH approximation form of the Laplacian of pressure and the divergence of intermediate velocity in Eq. (14) can be expressed as follows,

$$\nabla^2 p(\mathbf{x}_i, t) \approx \langle \nabla^2 p_i \rangle = \frac{2}{\bar{\rho}} \sum_j m_j \left( \frac{\mathbf{r}_{ij} \cdot \nabla W(r_{ij}, h)}{r_{ij}^2 + \eta^2} \right) (p_i - p_j), \quad (15)$$

$$\nabla \cdot \mathbf{v}(\mathbf{x}_i, t) \approx \langle \nabla \cdot \mathbf{v}_i \rangle = \frac{1}{\bar{\rho}} \sum_j m_j (\mathbf{v}_j - \mathbf{v}_i) \cdot \nabla W(r_{ij}, h). \quad (16)$$

Whereas the SPH particle density used in Eq. (14) is commonly evaluated by:

$$\rho(\mathbf{x}_i, t) \approx \langle \rho_i \rangle = \sum_j m_j W(r_{ij}, h). \quad (17)$$

#### 2.5. Modeling of turbulence stress

The turbulent stress term  $\boldsymbol{\tau}$  in (4) can be modeled by a large eddy simulation (LES) approach [40,41]. One can write the following expression of the turbulent stress:

$$\frac{\boldsymbol{\tau}}{\bar{\rho}} = 2\nu_T \mathbf{S} - \frac{2}{3}k \mathbf{1}, \quad (18)$$

where  $\nu_T$  and  $k$  are the turbulence eddy viscosity and the turbulence kinetic energy, respectively.  $\mathbf{S}$  indicates the strain rate tensor of the mean flow and  $\mathbf{1}$  is the identity tensor. Here, the isotropic term of Eq. (18) is normally appended in the pressure term to be solved in the PPE.

In the current work, the eddy viscosity is modeled by the static Smagorinsky model as,

$$\nu_T = (C_s \Delta l)^2 \|\bar{\mathbf{S}}\|, \quad (19)$$

in which the constant  $\Delta l$  is taken as  $2h$  in the SPH method, whereas  $C_s$  is the Smagorinsky constant whose value ranges from 0.1 to around 0.2 and normally requires adjustment for different simulation problems under different flow condition. The local strain rate can be calculated as  $\|\bar{\mathbf{S}}\| = \sqrt{2\mathbf{S}:\mathbf{S}}$ . In the current work, the averaged scalar rate of strain for each particle  $i$  can be calculated by the SPH approximated as [42],

$$\|\bar{\mathbf{S}}\|^2 (\mathbf{x}_i, t) \approx \left\langle \|\bar{\mathbf{S}}\|_i^2 \right\rangle = \frac{1}{2} \sum_j m_j \frac{\rho_i + \rho_j}{\rho_i \rho_j} \frac{\|\mathbf{v}_i - \mathbf{v}_j\|^2}{r_{ij}^2 + \eta^2} \mathbf{r}_{ij} \cdot \nabla W(r_{ij}, h). \quad (20)$$

## 2.6. Treatment of boundary conditions

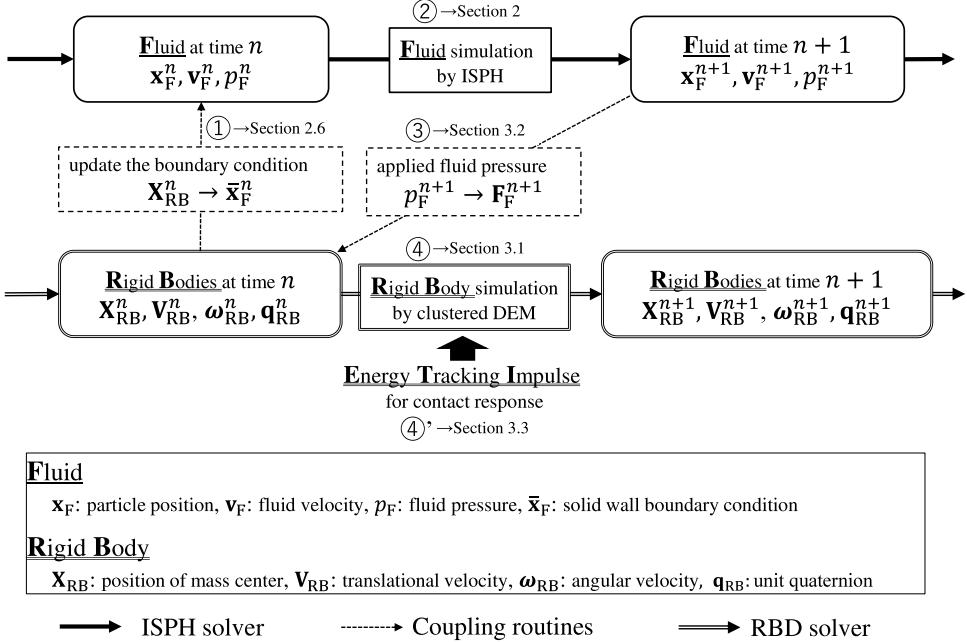
The treatment of free surface and impenetrability condition on the solid walls has an important role to obtain accurate pressure profiles while solving the stabilized PPE (14). In the current work, a simple free-surface detection algorithm proposed by [43] was implemented to enforce zero Dirichlet boundary conditions of the PPE. Meanwhile, a modified mirror boundary treatment with a virtual marker [44] is used to satisfy a reasonable slip or no-slip velocity field and accurate Neumann boundary condition of pressure at the vicinity of the wall boundaries.

## 3. Fluid–rigid-body interaction in a partitioned two-way coupling scheme

The frontier work of the fluid–rigid-body interaction using particle-based methods was pioneered by Koshizuka et al. [45]. They proposed a scheme which passively moves the rigid body and solves both of the fluid and rigid particles similarly, treating the rigid object as an incompressible material and applying additional rigid-body constraints on the solid particles. Other more recent methods, e.g. [19,20,46,47], simulate fluid–structure interaction problems commonly in a two-way coupling scheme, either in weakly or strongly-coupled assumption. As far as we have tested the above two approaches, the passively moving solid approach exhibits an inaccurate rotational motion during the rigid-body dynamics update, which is mainly rooted from the violation of dynamic equilibrium of rigid-body angular momentum and improper consideration of moment of inertia tensor. Meanwhile, the typical SPH–DEM approach is observed to be not so stable and accurate, particularly if multiple-body contact is involved. These inaccuracy and instability are mainly caused by the inherent nature of DEM contact assumption, which approximates collision response by artificial contact parameters as explained earlier.

The framework of the proposed coupling scheme for the fluid–rigid-body interaction simulator is similar to the one suggested in the SPH–DCDEM method [19], except for the rigid-body contact response routine, which is implemented by using the ETI approach proposed by Li et al. [31] for particle-discretized rigid-body. Although the SPH–DCDEM used a WCSPH approach and our proposed method utilizes the stabilized ISPH [36] to solve the fluid governing equations, we would like to emphasize that the main difference between the two formulations focused on the treatment of contact between rigid-bodies. Fig. 2 explains the main four processes of the two-way Neumann–Dirichlet weakly coupling scheme denoted as ①–④. The aim of the current paper is to introduce the ETI method to the overall FSI coupling scheme to provide a significantly more accurate estimation of contact response between multiple rigid objects.

In the current work, the rigid bodies are discretized into a finite number of particles with a constant particle mass  $m$  and a constant particle diameter  $d$ . The diameter  $d$  is set to be the same as the initial fluid particle distance in the ISPH. Here, the constant particle diameter is preferred as the rigid-body particles will act as a moving boundary, which is treated as a constraint (Dirichlet boundary condition) on the ISPH simulation. In a staggered weakly coupling assumption, the ISPH solves the governing equation on the fluid domain by considering the moving boundary condition obtained from the previous time step solution of the rigid bodies. Following that, the calculated fluid pressure from the Poisson equation is converted to external forces (Neumann boundary condition) acting on each rigid body surface particle within the DEM solver. This hydrodynamics force along with the gravitational force is applied to all rigid bodies prior to computing the contact response between these bodies and surrounding wall boundaries using the ETI method. Last but not least, the velocity, position, and orientation of the rigid bodies can be updated and the next time step can proceed.



**Fig. 2.** Two-way fluid–rigid-body coupling scheme in staggered manner.

### 3.1. Rigid body dynamics using accurate quaternion integration

The Newton–Euler equation which describes the dynamic equilibrium of linear and angular momentum of a rigid body can be written with the body's mass  $M$  and inertia tensor  $\mathbf{I}$  with respect to the center of mass  $\mathbf{X}$  as

$$M \frac{d\mathbf{V}}{dt} = \mathbf{F}, \quad (21)$$

$$\mathbf{I} \cdot \frac{d\boldsymbol{\omega}}{dt} + \boldsymbol{\omega} \times (\mathbf{I} \cdot \boldsymbol{\omega}) = \mathbf{T}, \quad (22)$$

where  $\mathbf{V} = \frac{d\mathbf{X}}{dt}$  and  $\boldsymbol{\omega}$  are the translational and angular velocities, respectively. Following the discretization of the rigid body into particles, the total force and moment,  $\mathbf{F}$  and  $\mathbf{T}$ , can be calculated considering the effect of gravity, hydrodynamic forces, and contact forces between rigid bodies or with a fixed boundary as follows;

$$\mathbf{F} = \mathbf{F}_F + \mathbf{F}_{RB} + Mg \quad (23)$$

$$\mathbf{T} = \mathbf{T}_F + \mathbf{T}_{RB} \quad (24)$$

where the subscripts F and RB indicate the fluid and rigid-body contact forces and/or moments.

In our particle-discretized rigid body simulation, the external force  $\mathbf{F}_\kappa(t)$  for component  $\kappa = \{F, RB\}$  acting on the body at any time  $t$  is assumed to be the summation of the same forces acting on each particle  $\mathbf{f}_{\kappa,i}(t)$  as,

$$\mathbf{F}_\kappa(t) = \sum_{i=1}^{n_p} \mathbf{f}_{\kappa,i}(t), \quad (25)$$

while the external moment is defined similarly as,

$$\mathbf{T}_\kappa(t) = \sum_{i=1}^{n_p} (\mathbf{x}_i(t) - \mathbf{X}(t)) \times \mathbf{f}_{\kappa,i}(t). \quad (26)$$

Here,  $n_p$  denotes the total number of particles constructing a rigid body, whereas  $\mathbf{x}_i$  indicates the position of rigid body particle  $i$ . The computation of the fluid force  $\mathbf{f}_{F,i}(t)$  and contact force  $\mathbf{f}_{RB,i}(t)$  in the particle level will be elaborated in Sections 3.2 and 3.3, respectively.

Having the translational and angular velocities in hand by solving Eqs. (21) and (22), the velocity of rigid body particles can be updated correspondingly. The velocity  $\mathbf{v}_i$  of rigid body particle  $i$  is obtained simply by combining the translational and angular velocities as

$$\mathbf{v}_i^{n+1} = \mathbf{V}^{n+1} + \boldsymbol{\omega}^{n+1} \times \mathbf{r}_i^{n+1}. \quad (27)$$

Here, the relative position vector  $\mathbf{r}_i$  at  $t = n + 1$  can be computed by subtracting the updated particle coordinate with the rigid body center as

$$\mathbf{r}_i^{n+1} = \mathbf{x}_i^{n+1} - \mathbf{X}^{n+1}. \quad (28)$$

These two position vectors are at the moment unknown and need to be obtained. Notice that, the body's translational motion will not change the particle's relative position  $\mathbf{r}_i$ , and if a rotation matrix  $\mathbf{R}$  or its increment  $d\mathbf{R}$  can be calculated, we can rewrite Eq. (28) as,

$$\mathbf{r}_i^{n+1} = \mathbf{R}^{n+1} \mathbf{r}_i^0 = d\mathbf{R}^{n+1} \mathbf{r}_i^n. \quad (29)$$

In order to locate a rigid body particle in the world space accurately, an appropriate understanding of spatial variables that represent the position and orientation of the rigid body is extremely important. The new position of the center of mass  $\mathbf{X}^{n+1}$  can be integrated from the obtained linear velocity  $\mathbf{V}^{n+1}$  by the following integration,

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \Delta t \mathbf{V}^{n+1}. \quad (30)$$

While the position of the rigid body can be updated through the simplectic Euler integration, the update of the body's orientation in 3D space is not straightforward, particularly, as the three-dimensional rotation over the body's local axis exhibits non-commutative and non-holonomic properties. Three main frameworks to describe the rotation of a rigid body's orientation are available in the literature: Euler angles, rotation matrices, and unit quaternion, all of which have advantages and disadvantages.

In the current work, unit quaternion is used and preferred as it overcomes the Gimbal lock [48] and the singularity problems [49] arise from the Euler angles and rotation matrices approach, respectively [50,51]. The integration process of orientation starts by converting the rotational velocity  $\boldsymbol{\omega}^{n+1}$  obtained from (22) to a unit quaternion  $\mathbf{q}^{n+1}$  following the process explained in the [Appendix](#). Following that, a rotation matrix  $\mathbf{R}^{n+1}$  can be constructed from the updated  $\mathbf{q}^{n+1}$  as,

$$\mathbf{R} = \begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}. \quad (31)$$

Notice that, by converting the unit quaternion to the corresponding rotation matrices, the rotation of rigid body orientation can be computed in a much less complicated way, eliminating the more arduous quaternion product, i.e. Eqs. (A.8) and (A.9). Having the rotation matrix in hand, one can compute other vectorial and tensorial quantities, such as the normal vector  $\mathbf{n}$  and the moment of inertia tensor  $\mathbf{I}$ , in the rotated frame of reference as:

$$\mathbf{n}_i^{n+1} = \mathbf{R}^{n+1} \mathbf{n}_i^0, \quad (32)$$

$$\mathbf{I}^{n+1} = \mathbf{R}^{n+1} \mathbf{I}^0 (\mathbf{R}^{n+1})^T. \quad (33)$$

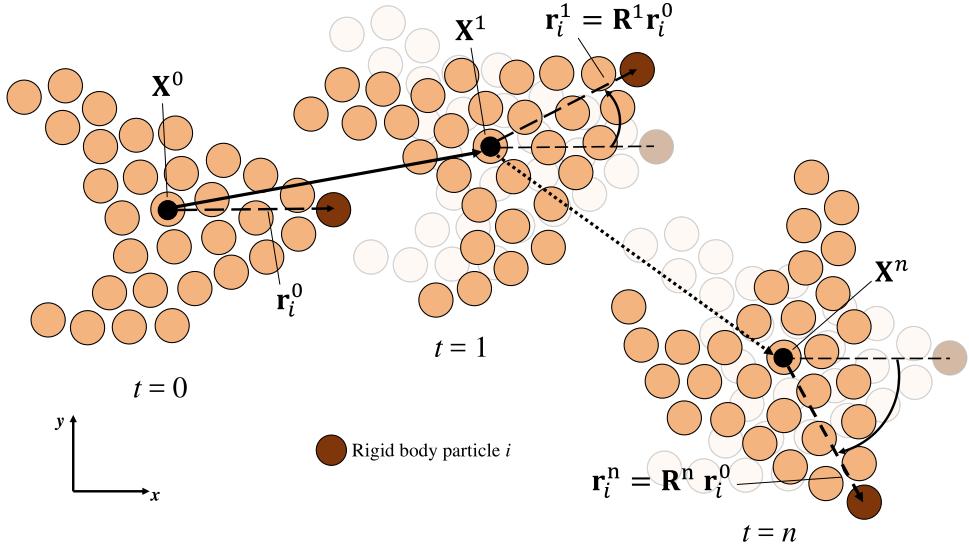
Finally, the location of rigid body particle  $i$  in world space  $\mathbf{x}_i$  is the result of first translating the body's center and then rotating the particle's relative position  $\mathbf{r}_i^0$  about the center of mass as,

$$\mathbf{x}_i^{n+1} = \mathbf{X}^{n+1} + \mathbf{r}_i^{n+1} = \mathbf{X}^{n+1} + \mathbf{R}^{n+1} \mathbf{r}_i^0. \quad (34)$$

[Fig. 3](#) illustrates the motion of rigid body particle  $i$  in space as the body translates and rotates over time.

### 3.2. Assessment of hydrodynamic force and moment

As a rigid body transiently moving throughout the spatial domain, the rigid-body particles constructing the body act as moving boundary conditions which should be considered and enforced in the ISPH calculations. At the beginning of every time step, the fluid solver is computed before the rigid-body solver. Here, the velocity  $\mathbf{v}_i^n$  of a



**Fig. 3.** Updating the position of rigid body particle  $i$  using the body's position and orientation,  $\mathbf{X}(t)$  and  $\mathbf{R}(t)$ .

rigid body particle  $i$  at the previous time step  $n$ , which was evaluated by Eq. (27), is considered to evaluate the pressure Neumann condition while solving the pressure Poisson equation.

The pressure Neumann condition should be enforced exactly on the rigid-body surface such that the interpenetration constraint is achieved correctly along the fluid-rigid boundary. Since there are no rigid body particles exactly located on the rigid body surface  $\Gamma_{\text{RB}}$ , first the coordinates of the rigid body surface,  $\mathbf{x}_\gamma$ , are identified from the outermost layer of rigid body particles  $s$  as:

$$\mathbf{x}_\gamma = \mathbf{x}_s + r_s \mathbf{n}_\gamma, \quad (35)$$

where  $r_s$  is the closest positive distance value measured from  $\mathbf{x}_s$  to the exact rigid-body surface and  $\mathbf{n}_\gamma (\equiv \mathbf{n}_s)$  is the outward unit normal vector perpendicular to the surface. These two quantities are initially evaluated during the discretization process from CAD data and stored, as well as updated at each time step, particularly when the rigid-body position and orientation change. These surface coordinates and the outward normal vectors are not only useful to compute the hydrodynamic force but also to estimate the contact response which will be explained in the following section.

In our FSI scheme, the hydrodynamic force applied on the rigid body particles can be computed by satisfying the following pressure Neumann condition at the rigid body surface,

$$\frac{\partial p}{\partial \mathbf{n}_\gamma} = \rho (\mu \nabla^2 \mathbf{v} + \mathbf{g}) \cdot \mathbf{n}_\gamma. \quad (36)$$

Then, the pressure  $p_s$  on the outermost layer of rigid body particles  $s$  can be evaluated numerically as follows:

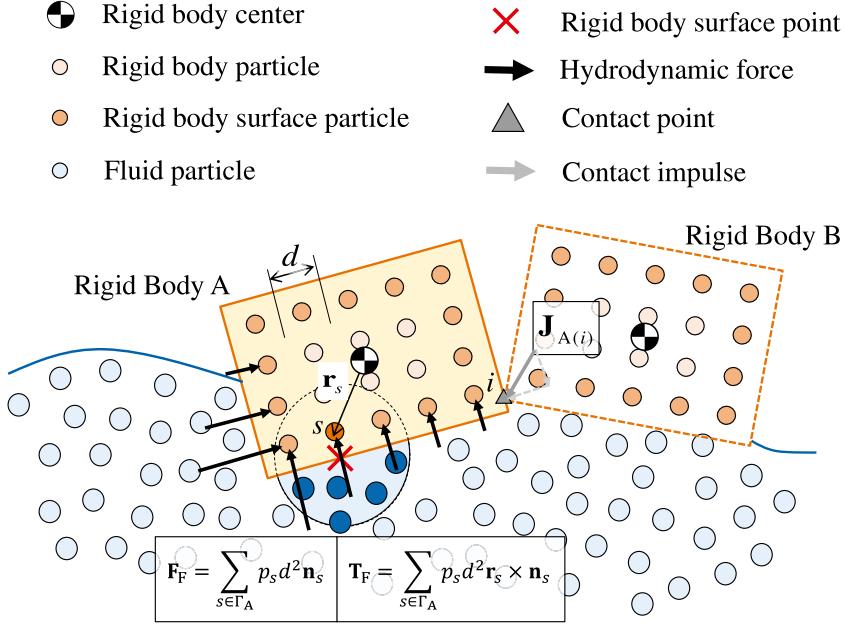
$$p_s = \langle p_\gamma \rangle - \|\mathbf{x}_s - \mathbf{x}_\gamma\| \rho (\mu \langle \nabla^2 \mathbf{v}_\gamma \rangle + \mathbf{g}) \cdot \mathbf{n}_\gamma, \quad (37)$$

where  $\langle p_\gamma \rangle$  denotes the fluid pressure on the rigid surface coordinate  $\gamma$ , or commonly called as the virtual marker by [44], which can be approximated through an SPH interpolation as:

$$\langle p_\gamma \rangle = \frac{\sum_{j \in \Omega_F} \frac{m_j}{\rho_j} p_j W(r_{\gamma j}, h)}{\sum_{k \in \Omega_F} W(r_{\gamma k}, h)} = \sum_{j \in \Omega_F} \frac{m_j}{\rho_j} p_j \tilde{W}(r_{\gamma j}, h), \quad (38)$$

$$\tilde{W}(r_{\gamma j}, h) := \frac{W(r_{\gamma j}, h)}{\sum_{k \in \Omega_F} W(r_{\gamma k}, h)}. \quad (39)$$

Here,  $\tilde{W}(r_{\gamma j}, h)$  is a normalized kernel function to ensure the partition of unity condition at the vicinity of the rigid body surface and subscript  $j$  denotes the surrounding fluid particle constructing the fluid volume  $\Omega_F$ . Following



**Fig. 4.** Measuring hydrodynamic forces by using a virtual marker at rigid-body surfaces.

that, the Laplacian velocity term  $\langle \nabla^2 \mathbf{v}_\gamma \rangle$  specified in Eq. (37) is given by:

$$\langle \nabla^2 \mathbf{v}_\gamma \rangle = \sum_{j \in \Omega_F \cup \Omega_{RB}} m_j \left( \frac{\rho_\gamma + \rho_j}{\rho_\gamma \rho_j} \frac{\mathbf{r}_{\gamma j} \cdot \nabla W(r_{\gamma j}, h)}{r_{\gamma j}^2 + \eta^2} \right) (\mathbf{v}_\gamma - \mathbf{v}_j) . \quad (40)$$

Note that, in Eqs. (38)–(40), the distance  $r_{\gamma j} = \|\mathbf{r}_{\gamma j}\| = \|\mathbf{x}_\gamma - \mathbf{x}_j\|$ , the density  $\rho_\gamma$  can be assumed equal to  $\rho_s$ , and the velocity of the surface point  $\mathbf{v}_\gamma$  can be obtained from:

$$\mathbf{v}_\gamma = \mathbf{v}_s + \boldsymbol{\omega} \times (\mathbf{x}_\gamma - \mathbf{x}_s) . \quad (41)$$

Fig. 4 shows a schematic image to evaluate the fluid forces on the rigid-body surface particles. The representative area for all the outermost layer of rigid body particles  $s$  is approximated by  $d^2$  for simplicity, and then, by considering the computed pressure from Eq. (37), the hydrodynamic force and the corresponding moment can be finally obtained by taking a summation over  $s$  following Eqs. (25) and (26) as:

$$\mathbf{F}_F = \sum_{s \in \Gamma_{RB}} \mathbf{f}_{F,s} = \sum_{s \in \Gamma_{RB}} p_s d^2 \mathbf{n}_s , \quad (42)$$

$$\mathbf{T}_F = \sum_{s \in \Gamma_{RB}} (\mathbf{x}_s - \mathbf{X}) \times \mathbf{f}_{F,s} = \sum_{s \in \Gamma_{RB}} p_s d^2 (\mathbf{x}_s - \mathbf{X}) \times \mathbf{n}_s . \quad (43)$$

### 3.3. Energy-tracking impulse (ETI) contact algorithm in an FSI simulation

In the proposed FSI scheme, the treatment of contact force is decoupled from the other external forces, e.g. hydrodynamics and body forces. First, the gravitational and fluid forces and their corresponding moments are applied to the rigid body, yielding intermediate translational and angular velocities,  $\mathbf{V}^*$  and  $\boldsymbol{\omega}^*$ , respectively:

$$M \frac{d\mathbf{V}^*}{dt} = \mathbf{F}_F^{n+1} + M\mathbf{g} , \quad (44)$$

$$\mathbf{I}^n \cdot \frac{d\boldsymbol{\omega}^*}{dt} + \boldsymbol{\omega}^n \times (\mathbf{I}^n \cdot \boldsymbol{\omega}^n) = \mathbf{T}_F^{n+1} . \quad (45)$$

Here, notice that the computed intermediate velocity fields are not used to update the position of the rigid body yet in this stage. This means that, if position and velocity constraint conditions between rigid bodies are violated, necessary contact impulses should be applied to avoid interpenetration of rigid-body surfaces.

The constraints on position and velocity between rigid body A and B are expressed as:

$$C_p(\mathbf{x}_{A(i)}^n, \mathbf{x}_{B(j)}^n) := (\mathbf{x}_{A(i)}^n - \mathbf{x}_{B(j)}^n) \cdot \mathbf{n}_{B(j)}^n \geq 0, \quad (46)$$

$$\begin{aligned} C_v(\mathbf{x}_{A(i)}^n, \mathbf{x}_{B(j)}^n) &:= (\mathbf{v}_{A(i)}^* - \mathbf{v}_{B(j)}^*) \cdot \mathbf{n}_{B(j)}^n \\ &= (\{\mathbf{V}_A^* + \boldsymbol{\omega}_A^* \times (\mathbf{x}_{A(i)}^n - \mathbf{X}_A^n)\} - \{\mathbf{V}_B^* + \boldsymbol{\omega}_B^* \times (\mathbf{x}_{B(j)}^n - \mathbf{X}_B^n)\}) \cdot \mathbf{n}_{B(j)}^n \geq 0, \end{aligned} \quad (47)$$

for  $\forall \mathbf{x}_{A(i)}$  on  $\Gamma_A$ , and  $\forall \mathbf{x}_{B(j)}$  on  $\Gamma_B$ .

The above constraint conditions should always be satisfied for all arbitrary contact points  $\mathbf{x}_{A(i)}^n$  and  $\mathbf{x}_{B(j)}^n$  on the surface of a “master” body  $\Gamma_A$  and a “slave” body  $\Gamma_B$ . If any points on body A and B violate the impenetrability constraint (Eq. (46)), the contact algorithm routine should be called to approximate equivalent contact forces that satisfy both the aforementioned position and velocity constraints. However, conversely, if the constraint conditions are satisfied by all points on the bodies, the next-step velocities can be simply determined from the intermediate values, i.e.  $\mathbf{V}^{n+1} = \mathbf{V}^*$  and  $\boldsymbol{\omega}^{n+1} = \boldsymbol{\omega}^*$ .

The ETI method conceptually assumes an iterative application of impulses with well-controlled energy storage, dissipation, and release consideration during contact. In the ETI approach, once the impulsive contact forces  $\mathbf{J}_{RB(i)}^k$  and moments  $\mathbf{r}_i \times \mathbf{J}_{RB(i)}^k$  are applied on any contacting particle  $i$  at iteration  $k$ , an incremental translational and an angular velocity,  $\Delta\mathbf{V}^k$  and  $\Delta\boldsymbol{\omega}^k$ , can be obtained from the following equations:

$$M \Delta\mathbf{V}^k = \mathbf{J}_{RB(i)}^k, \quad (48)$$

$$\mathbf{I}^n \cdot \Delta\boldsymbol{\omega}^k = \mathbf{r}_i^n \times \mathbf{J}_{RB(i)}^k. \quad (49)$$

The computed incremental velocities are then used to correct the predicted intermediate velocities,  $\mathbf{V}^*$  and  $\boldsymbol{\omega}^*$ , calculated by Eqs. (44) and (45). The next iteration translational and angular velocities at  $k+1$  can be computed as follows:

$$\mathbf{V}^{k+1} = \mathbf{V}^k + \Delta\mathbf{V}^k = \mathbf{V}^* + \sum_{l=1}^k \Delta\mathbf{V}^l, \quad (50)$$

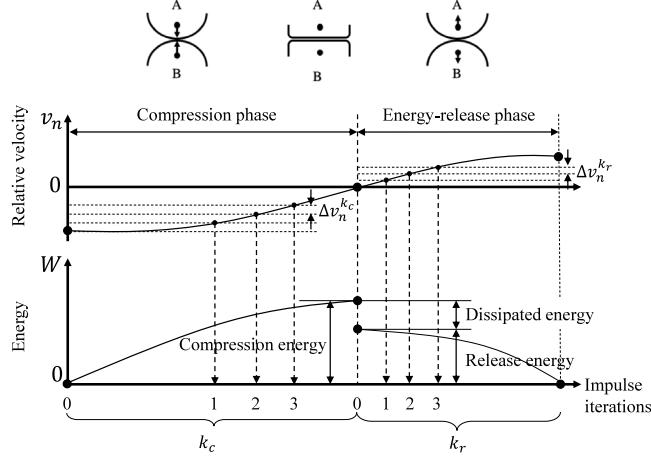
$$\boldsymbol{\omega}^{k+1} = \boldsymbol{\omega}^k + \Delta\boldsymbol{\omega}^k = \boldsymbol{\omega}^* + \sum_{l=1}^k \Delta\boldsymbol{\omega}^l. \quad (51)$$

At each time step, the ETI contact response computation is divided into two distinct phases: compression and release, as shown in Fig. 5. The compression phase absorbed the compression energy after applying incremental impulse at a contact point with a priority based on the minimum relative velocity. On the other hand, in the release phase, the stored compression energy is released gradually by applying impulse similarly to the first phase. However, in this phase, the sequence of impulse application is ordered from the contact point with the maximum remaining energy storage. For surface-to-surface contact, the impulse is applied to several contact points with approximately the same value of relative velocity and remaining energy for the compression and release phase, respectively. In the current work, in between the two phases, an energy dissipation process based on the Stronge’s hypothesis is considered by means of the coefficient of restitution. It is important to note that the velocity of the two contacting bodies is updated incrementally during the ETI impulse iterations, although the position and the orientation of the bodies should be updated only once after all the ETI iterations are completed.

### 3.3.1. ETI compression phase

In the compression phase, the incremental impulse is applied from the point which has the minimum value of normal relative velocity,  $v_{n_i}^{k_c} = (\mathbf{v}_{A(i)}^{k_c} - \mathbf{v}_{B(i)}^{k_c}) \cdot \mathbf{n}_i$ , within the global contact candidate points. The subscripts A and B denote the two contacting bodies. Then, an incremental relative velocity  $\Delta v_{n_i}^{k_c}$  with a scaling factor  $\lambda^{\text{comp}}$  should be estimated after identifying a point  $i$  with the minimum normal relative velocity at the compression phase iteration  $k_c$  following [31]. One can compute:

$$\Delta v_{n_i}^{k_c} = \frac{\min(v_{n_i}^{k_c})}{\lambda^{\text{comp}}}, \quad (52)$$



**Fig. 5.** ETI phases and impulse iteration. Here,  $k_c$  and  $k_r$  indicate the iteration counter for the compression and energy-release phase, respectively.

where the scaling factor  $\lambda^{\text{comp}}$  depends on the number of contact points with approximately the same relative velocity compared to  $\min(v_{n_i}^{k_c})$ . Here, we intend to compute an incremental normal impulse  $J_{n_i}^{k_c}$  such that, if it is applied to the contact point  $i$ , the incremental normal relative velocity  $\Delta v_{n_i}^{k_c}$  at the same contact point can be eliminated.

The energy due to collision is first absorbed assuming friction-less condition at contact point  $i$ . Here, the energy absorbed at compression iteration  $k_c$  is expressed as

$$\Delta W_i^{k_c, \text{comp}} = \frac{1}{2} \left( 2v_{n_i}^{k_c} + \Delta v_{n_i}^{k_c} \right) J_{n_i}^{k_c}. \quad (53)$$

The impulse along the normal direction  $J_{n_i}^{k_c}$  can be calculated as,

$$J_{n_i}^{k_c} = \frac{\Delta v_{n_i}^{k_c}}{\mathbf{n}_i^T \mathbf{K}_i \mathbf{n}_i}. \quad (54)$$

Here,  $\mathbf{n}_i$  is the unit normal vector at contact point  $i$  and  $\mathbf{K}_i$  is a  $3 \times 3$  collision matrix at contact point  $i$ , which can be obtained as,

$$\mathbf{K}_i = \left( \frac{1}{M_A} + \frac{1}{M_B} \right) \mathbf{1} - (\tilde{\mathbf{r}}_{A(i)} \mathbf{I}_A^{-1} \tilde{\mathbf{r}}_{A(i)} + \tilde{\mathbf{r}}_{B(i)} \mathbf{I}_B^{-1} \tilde{\mathbf{r}}_{B(i)}). \quad (55)$$

In the expression above,  $\mathbf{1}$  is the  $3 \times 3$  identity matrix, whereas  $\mathbf{I}$  denotes the inertia tensor. Here,  $\tilde{\mathbf{r}}_{RB(i)}$  is the distance of contact point  $i$  to the gravitational center of contacting bodies  $RB = \{A, B\}$ . The expression,  $\tilde{\mathbf{r}}_{RB(i)}$  is simply the same vector written in its cross-product matrix form as,

$$\tilde{\mathbf{r}} = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}. \quad (56)$$

When friction is considered, the friction impulse along the tangential direction of the relative motion at each contact point, that is,

$$J_{t_i}^{k_c} = \frac{|v_{t_i}^{k_c}|}{(\mathbf{t}_i^{k_c})^T \mathbf{K}_i \mathbf{t}_i^{k_c}}, \quad (57)$$

where  $v_{t_i}^{k_c} = (\mathbf{v}_{A(i)}^{k_c} - \mathbf{v}_{B(i)}^{k_c}) \cdot \mathbf{t}_i^{k_c}$  is the tangential relative velocity at contact point  $i$  and  $\mathbf{t}_i^{k_c}$  is the tangential unit vector at iteration  $k_c$  computed by:

$$\mathbf{t}_i^{k_c} = \frac{(\mathbf{v}_{A(i)}^{k_c} - \mathbf{v}_{B(i)}^{k_c}) - v_{n_i}^{k_c} \mathbf{n}_i}{\|(\mathbf{v}_{A(i)}^{k_c} - \mathbf{v}_{B(i)}^{k_c}) - v_{n_i}^{k_c} \mathbf{n}_i\|}. \quad (58)$$

Following the Coulomb's friction law, the maximum value of frictional impulse can be determined from the coefficient of friction  $\mu$  and the magnitude of the normal impulse as,  $\bar{J}_{t_i} = \min(J_{t_i}, \mu J_{n_i})$ . If the tangential impulse calculated from Eq. (57) is less than  $\mu J_n$ , the static frictional contact condition is assumed and the velocity can be updated by using Eqs. (54) and (57). The impulse vector during each compression phase can be written as:

$$\mathbf{J}_i^{k_c} = J_{n_i}^{k_c} \mathbf{n}_i + J_{t_i}^{k_c} \mathbf{t}_i^{k_c}. \quad (59)$$

Conversely, if the computed tangential impulse is larger than  $\mu J_n$ , the dynamic friction condition takes place, and the normal impulse needs to be modified before updating the velocities of each rigid body. We write the modified normal and tangential impulses, denoted with superscript +, as:

$$J_{n_i}^{k_c,+} = \frac{\Delta v_{n_i}^{k_c}}{\mathbf{n}_i^T \mathbf{K}_i (\mathbf{n}_i + \mu \mathbf{t}_i^{k_c})}, \quad (60)$$

$$J_{t_i}^{k_c,+} = \mu J_{n_i}^{k_c,+}. \quad (61)$$

The compression-phase impulse vector (59) is then modified as,

$$\mathbf{J}_i^{k_c} = (J_{n_i}^{k_c,+} - J_{n_i}^{k_c}) \mathbf{n}_i + J_{t_i}^{k_c,+} \mathbf{t}_i^{k_c}. \quad (62)$$

In each iteration of the compression phase, the translational and angular velocities can be updated by Eqs. (50) and (51) using the increments given by Eqs. (48) and (49).

### 3.3.2. Energy dissipation during contact

After all the contact points reach the maximum compression, in which there is no negative relative velocity remaining, the next step is to apply the Stronge's hypothesis [29] to dissipate the total compression energy  $W_i^{\text{comp}} = \sum_{k_c} \Delta W_i^{k_c, \text{comp}}$  during contact. One can write,

$$W_i^{\text{release}} = \epsilon^2 W_i^{\text{comp}}, \quad (63)$$

where  $\epsilon$  denotes the coefficient of restitution, whose value lies in the range of  $0 \leq \epsilon \leq 1$ . Meanwhile,  $W_i^{\text{release}}$  indicates the energy to be released during the release phase.

### 3.3.3. ETI release phase

In the release phase, the remaining compression energy after the dissipation process is released in a similar manner. The maximum remaining energy determines the priority queue of the released phase instead of the minimum relative velocity as in the compression phase. Let us denote the contact point with maximum energy with subscript  $m$  from hereon. The changes in velocity and its corresponding impulse are applied gradually similar to the iterative process done in the compression phase. After knowing the contact point  $m$  with the highest release energy, the maximum change allowed in normal velocity in a contact point  $i$  at a release iteration  $k_r$  can be computed as,

$$\Delta v_{n_i}^{k_r} = \frac{\Delta v_n^{\text{lim}}}{\lambda^{\text{release}}}, \quad (64)$$

where  $\lambda^{\text{release}}$  is a divider for the energy-release phase and,

$$\Delta v_n^{\text{lim}} = -v_{nm}^{k_r} + \sqrt{\left(v_{nm}^{k_r}\right)^2 + 2W_m^{k_r} (\mathbf{n}_m^T \mathbf{K}_m \mathbf{n}_m)}. \quad (65)$$

Following this process, the normal and tangential impulses can be computed in a similar way as in the compression phase, i.e. Eqs. (54)–(62), by using  $\Delta v_{n_i}^{k_r}$  instead of  $\Delta v_{n_i}^{k_c}$ , and thus, all the superscripts  $k_c$  should be changed to  $k_r$  in expressions (54)–(62). Once the impulse in the energy-release phase is computed, the subsequent energy released  $\Delta W_i^{k_r, \text{release}}$  can be evaluated in a similar way to Eq. (53). The energy-release phase iteration proceeds until all the remaining stored energy at all contact points are released totally.

To sum up, the total procedure of the proposed FSI formulation using the ISPH and cluster DEM methods with ETI contact is summarized in Fig. 6. The reader is strongly recommended to refer to [31] for more detailed information on the ETI algorithm.

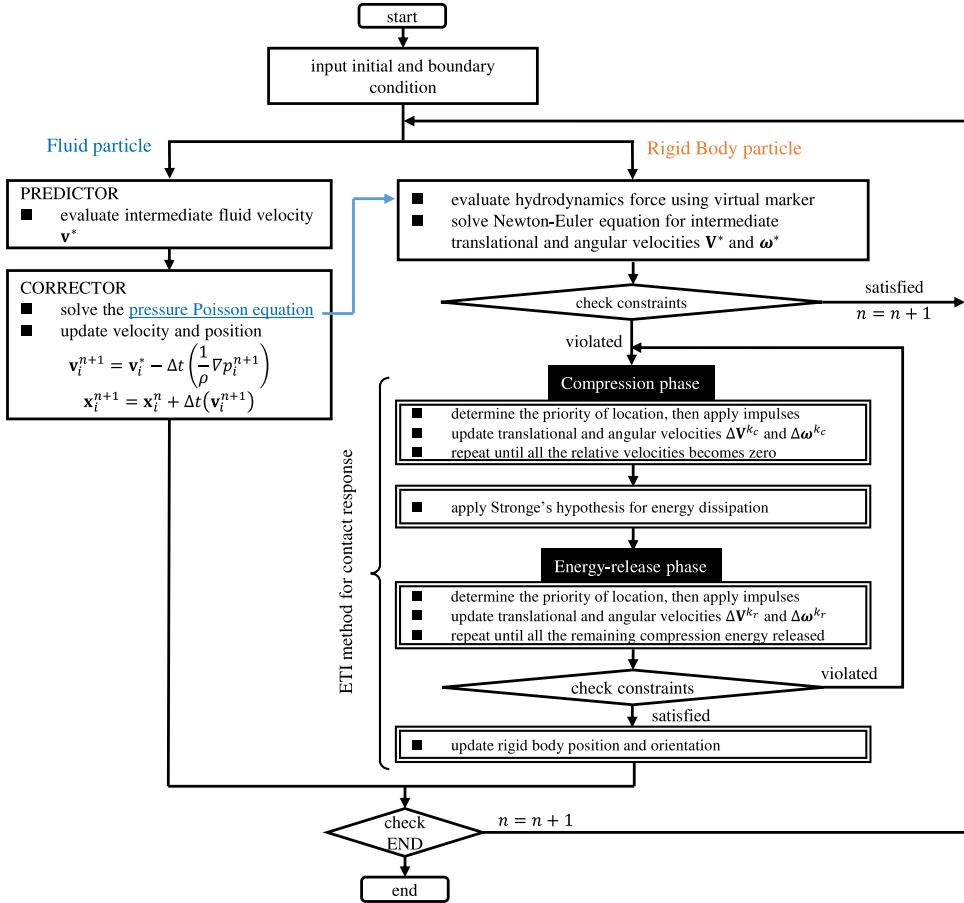


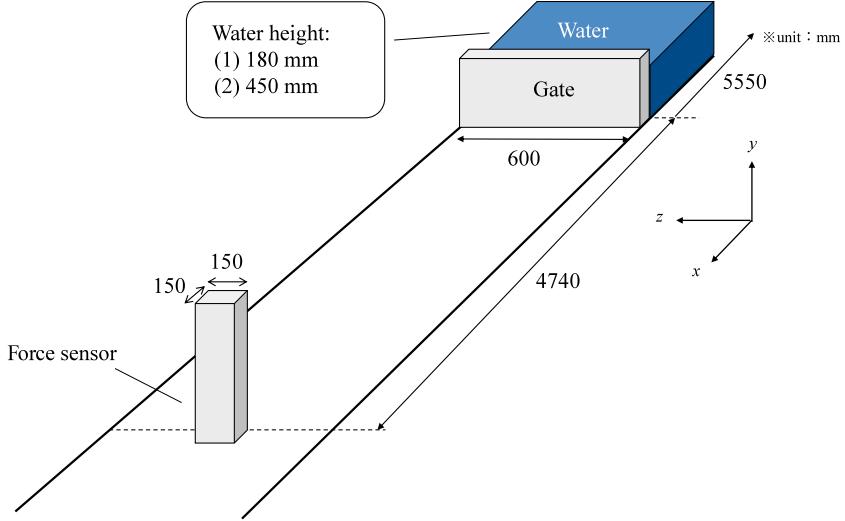
Fig. 6. Fluid–rigid-body interaction simulation flowchart with ETI contact response.

#### 4. Validation tests and numerical examples

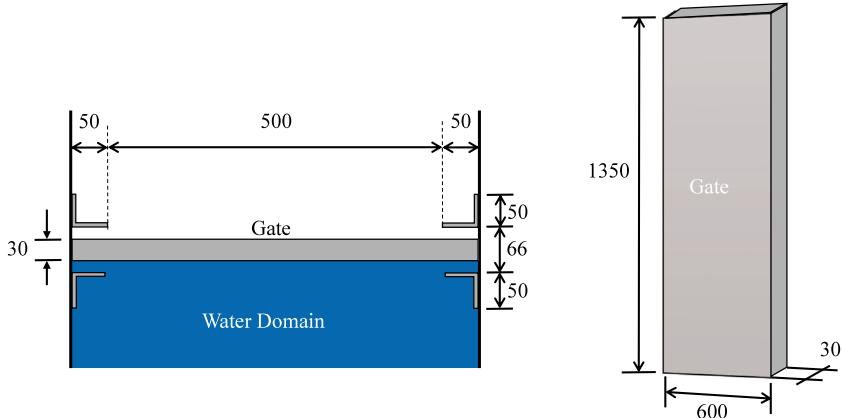
This section aims at assessing the quality of the numerical solutions achieved with the proposed ISPH–DEM FSI formulation by incorporating the ETI method to approximate the contact response. Four numerical tests have been studied and will be presented in the following sections:

1. A comparison study between our ISPH fluid solver and a dam-break experiment to assess the accuracy of the fluid solver and the method to approximate the hydrodynamic force on a solid boundary.
2. A bridge-washout simulation to analyze the accuracy of the proposed FSI coupling scheme for one rigid body. The obtained numerical results were compared with several experiments with varying initial water height. The numerical solutions obtained by using the ETI are also compared with numerical solutions given by the penalty method.
3. An experimental validation of fluid interacting with multiple rigid blocks with different arrangements to confirm the capability of the ETI contact algorithm to simulate the impact and collision of multiple objects.
4. A numerical test to elaborate the robustness and performance of the proposed FSI scheme for arbitrary-shaped rigid bodies.

For all cases, the dynamic viscosity and the material density of water are considered to be  $\mu_w = 8.9 \times 10^{-4}$  Pa·s and  $\rho_w = 1000$  kg/m<sup>3</sup>, respectively. The SPH smoothing length was set to be  $h = 1.2d$ , whereas the relaxation coefficient  $\alpha$  to stabilize the PPE in the ISPH, Eq. (14), was set at 0.01 for all simulations. Moreover, a hybrid slip and non-slip condition [44] is considered with the ratio of 2:1 for all simulations to obtain an accurate velocity profile normal to the solid wall.



**Fig. 7.** Hydrodynamic force measurement: experimental model of the water channel and the force sensor (image not to scale).



**Fig. 8.** Hydrodynamic force measurement: detailed dimensions of the dam-break gate (image not to scale).

#### 4.1. Measurement of hydrodynamic force

An experiment to measure the hydrodynamic force of a dam break was conducted as a preliminary test to check the accuracy of the SPH fluid solver and the force measurement technique numerically. A schematic figure of the considered model used in the experiment and the simulation is shown in Fig. 7, whereas the details of the opening gate are illustrated in Fig. 8. The experiment was conducted at a 16 m-long, 0.6 m-width wave channel located at the Hachinohe Institute of Technology in Aomori, Japan. The experiments were carried out three times for two different cases of initial water height  $H$ : 180 mm and 450 mm, and the force sensor was set at the position around 4.74 m ahead of the dam-break gate.

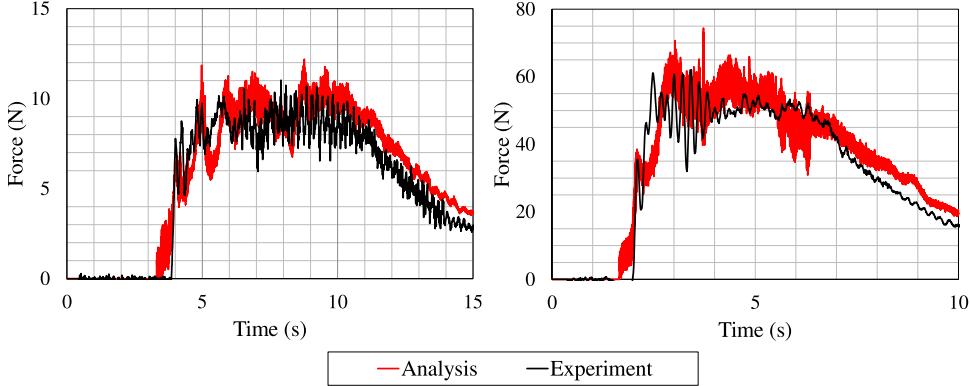
Table 1 summarizes the used numerical parameters. The simulation models were built up by particles with a diameter  $d = 0.5$  cm, totaling up to 13.2 and 20.4 million particles for  $H = 180$  and 450 mm, respectively. The Smagorinsky constant for the turbulence LES model was considered to be  $C_s = 0.1$  and the time step size is set as 0.001 s. In the current test, the hydrodynamic force of the dam-break flow in a channel is measured on the front surface of the installed force sensor.

The averaged value of the experiment data is used for comparison with the simulation results as shown in Fig. 9. In general, the results obtained showed good agreement with each other. As one can notice, there is a slight difference

**Table 1**

Hydrodynamic force measurement: simulation parameters considered.

|                                      |                               |
|--------------------------------------|-------------------------------|
| Particle diameter ( $d$ )            | 0.5 cm                        |
| Time step ( $\Delta t$ )             | 0.001 s                       |
| Fluid:                               |                               |
| Density ( $\rho_w$ )                 | 1000 kg/m <sup>3</sup>        |
| Dynamic viscosity ( $\mu_w$ )        | 8.9 × 10 <sup>-4</sup> Pa · s |
| Relaxation coefficient ( $\alpha$ )  | 0.01                          |
| Smagorinsky constant ( $C_s$ )       | 0.1                           |
| Ratio of slip and non-slip condition | 2:1                           |



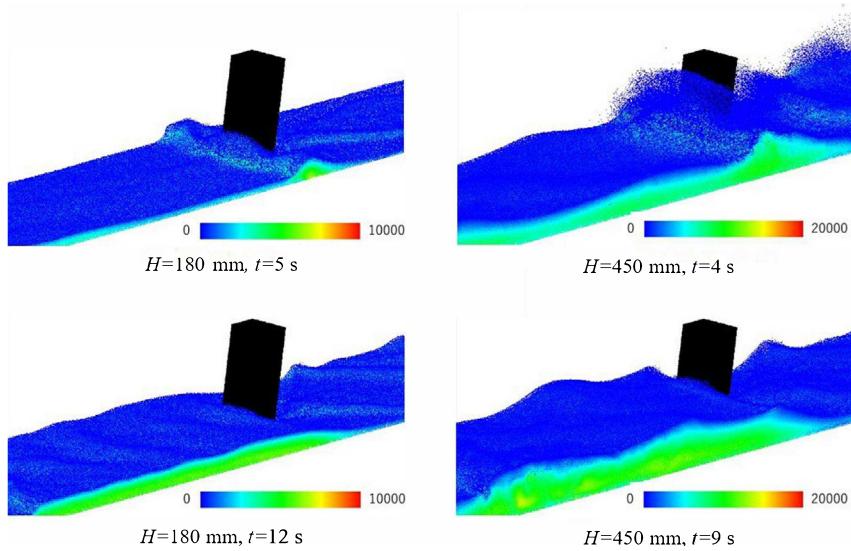
**Fig. 9.** Hydrodynamic force measurement: comparison of measured hydrodynamic force for two initial water height cases: (left)  $H = 180$  mm and (right)  $H = 450$  mm.

in the starting time, when the water first impacts the force sensor, between the numerical results and experimental data. This is caused mainly by the particle resolution and the method used to measure the pressure on the rigid body surface particle. As mentioned in Section 3.2, the virtual marker technique is used to interpolate the fluid pressure field around the rigid surface particles. This means that the hydrodynamics force will be recorded as soon as there is a fluid particle entering the support domain of these virtual markers, even though the fluid particles are not physically touching the force sensor yet. This, hence, causing a slight numerical error, however, should reduce along with particle refinement [36]. It should be noted that, even though the initial hydrodynamic force of the simulation results has some errors, the entire force profile in both the 180 mm and 450 mm cases shows very good agreement with the experiment. The simulation results for the two cases are visualized at selected time instance in Fig. 10.

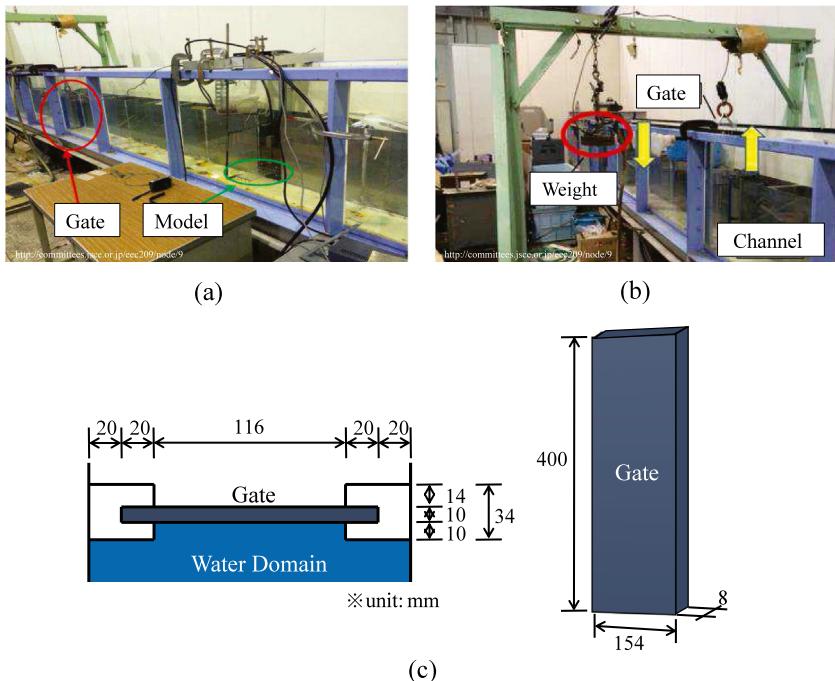
#### 4.2. Bridge washout simulation and experimental validation

Following the accurate measurement of hydrodynamic forces shown in Section 4.1, an experimental validation was conducted to check the robustness of the proposed method to handle fluid–solid coupled problems. Previously, our rigid-body contact algorithm has been profoundly proved to work correctly in both friction and frictionless collision contact [31]. To validate the interaction between the two physics, i.e. fluid and solid, a three-dimensional comparison between numerical solutions with experimental results is presented.

The wave channel of the Earthquake Engineering Laboratory at Ritsumeikan University, Shiga, Japan (Fig. 11), was adopted to perform a simulation of a simplified box-girder bridge subjected to a dam-break. The flume was sectioned at 6.0 m long and is 0.20 m wide, with glass side walls to grant optical access to the flow. The flume side and bottom walls can be considered to be very smooth. A gate was installed to hold the dam-break initially, with a cross-section and detailed dimension described by Fig. 11(c). Together with a lock lever and a monolithic weight, the pulley system allows for an ‘instantaneous’ opening action and removal of the dam-break.

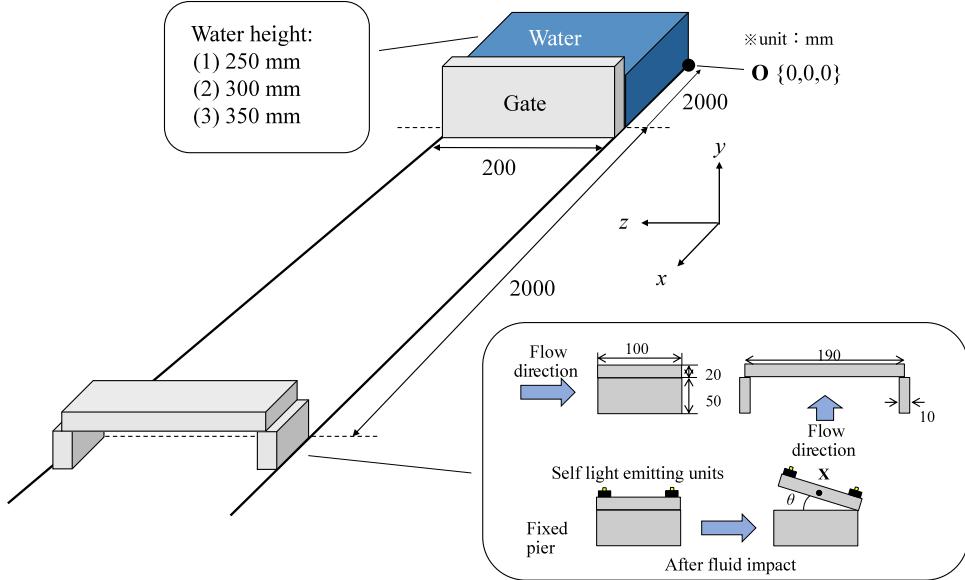


**Fig. 10.** Hydrodynamic force measurement: visualization of the conducted simulations: (left)  $H = 180$  mm at 5 s and 12 s and (right)  $H = 450$  mm at 4 s and 9 s. Color contour plots pressure measured in  $\text{g cm}^{-1} \text{s}^{-2} \equiv 0.1 \text{ Pa}$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 11.** Bridge washout: (a & b) wave channel where the experimental tests were conducted; (c) the schematic illustration of the opening gate (image not to scale).

In our experiments, Nobby Tech's VENUS 3-D real-time motion measurement system based on motion capture technology was utilized to observe the detailed movements of the bridge-girder during washout. Six synchronized motion capture cameras were set around the experimental field to track the bridge-girder movement as the dam break flow progresses. In this system, self-light emitting units are attached to the four corners and the upper center



**Fig. 12.** Bridge washout: schematic illustration of the analysis model (image not to scale).

**Table 2**

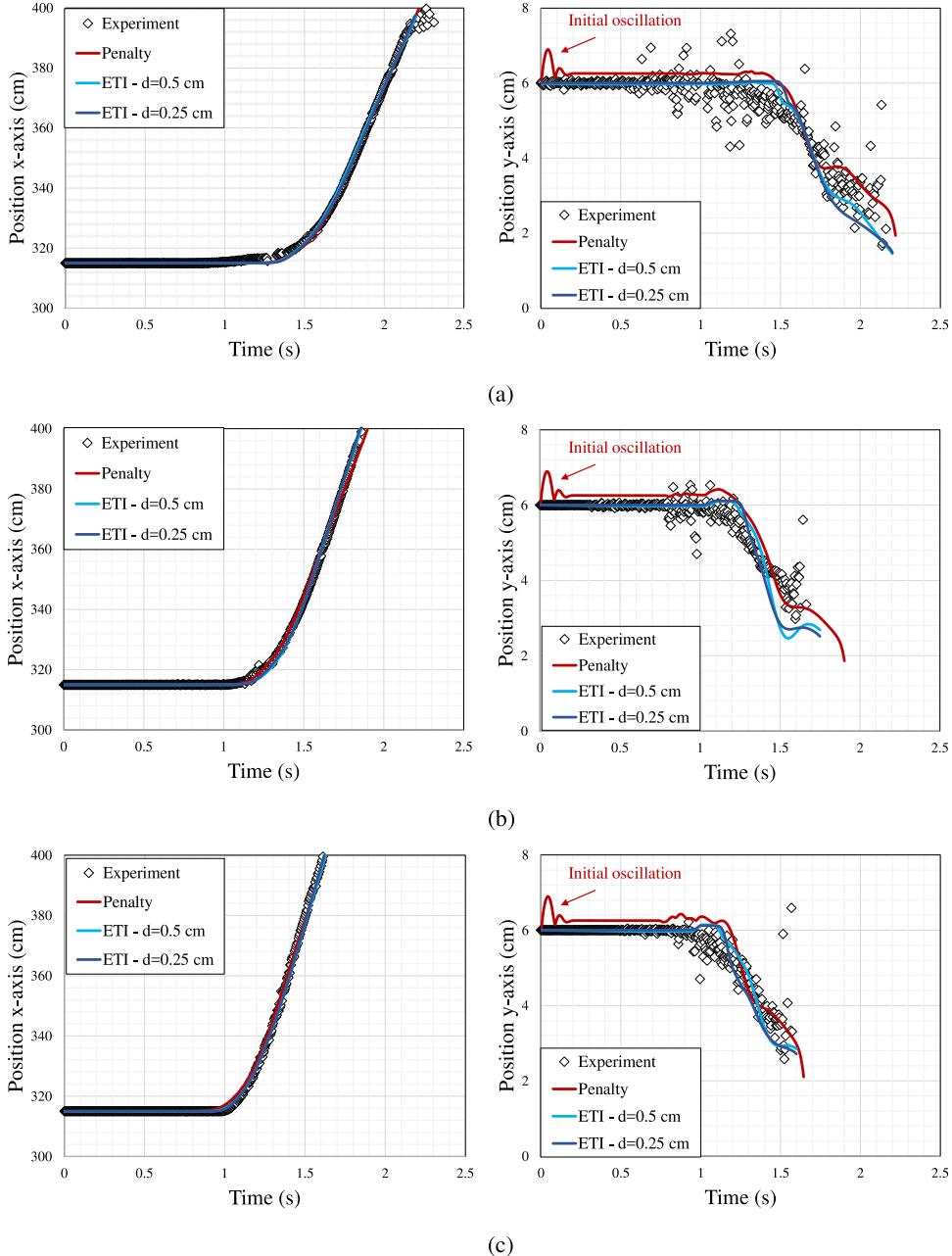
Bridge washout: simulation parameters considered.

|                                      |                                       |
|--------------------------------------|---------------------------------------|
| Particle diameter ( $d$ )            | (i) 0.5 cm; (ii) 0.25 cm              |
| Time step ( $\Delta t$ )             | 0.0005 s                              |
| Fluid:                               |                                       |
| Density ( $\rho_w$ )                 | 1000 kg/m <sup>3</sup>                |
| Dynamic viscosity ( $\mu_w$ )        | $8.9 \times 10^{-4}$ Pa · s           |
| Relaxation coefficient ( $\alpha$ )  | 0.01                                  |
| Smagorinsky constant ( $C_s$ )       | 0.2                                   |
| Ratio of slip and non-slip condition | 2:1                                   |
| Bridge-girder:                       |                                       |
| Density ( $\rho$ )                   | 1161 kg/m <sup>3</sup>                |
| Friction coeff. ( $\mu$ )            | 0.05 (girder-pier); 0.0 (girder-wall) |
| Coeff. restitution ( $\epsilon$ )    | 0.0 (rest); 0.8 (collision)           |

of the bridge model, and the location of the bridge-girder can be measured and tracked by using the installed cameras.

The experimental tests were carried out three to four times in three different cases of initial water height  $H$ : 250, 300, and 350 mm. Upon receiving the hydrodynamics impact from the fluid flow, the location of the girder model is monitored and recorded by using the aforementioned motion capture systems. The bridge piers were fixed, while the upper girder was placed above the piers and subjected to the dam-break. The density of the girder is measured to be  $\rho = 1161 \text{ kg/m}^3$ , referring to the density of PLA filament, the material used to make the experimental model. The details of the analysis model are shown in Fig. 12.

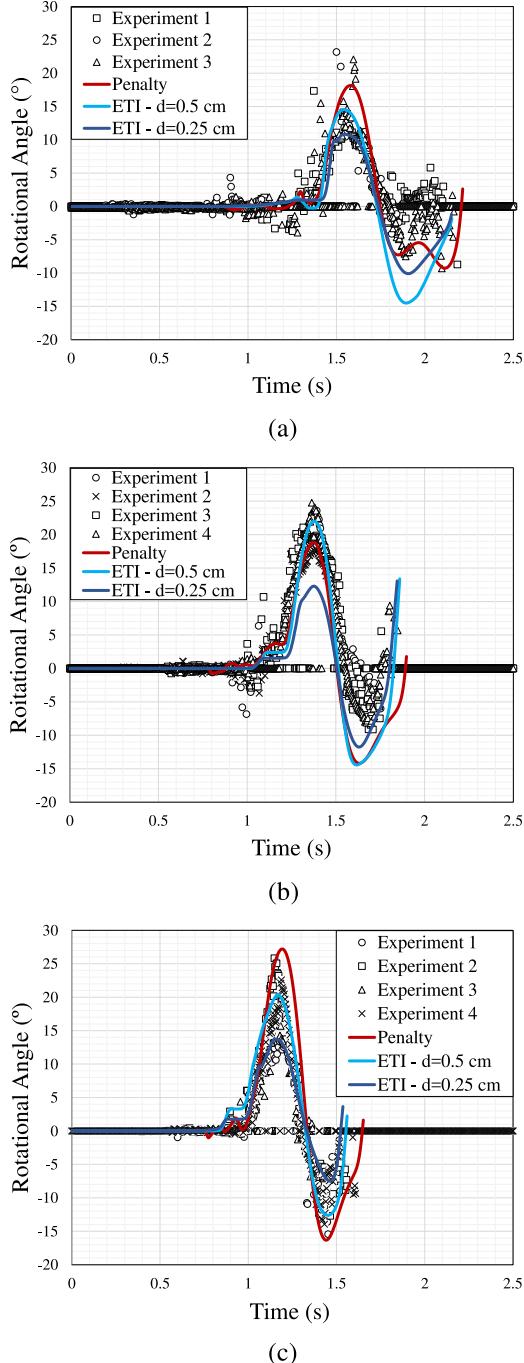
In the numerical analysis, the simulation parameters considered are summarized in Table 2. Two different sizes of particle diameter are assumed, i.e.  $d = 0.5 \text{ cm}$  and  $d = 0.25 \text{ cm}$ . The total number of particles for  $d = 0.5 \text{ cm}$  is about 2.4, 2.6, and 2.7 million, whereas for  $d = 0.25 \text{ cm}$ , it is about 12.6, 13.9, and 15.2 million, for  $H = 250, 300$ , and 350 mm, respectively. The Smagorinsky constant was considered to be  $C_s = 0.2$ . The material parameters of the rigid body considered to formulate the contact force are the coefficient of friction  $\mu$  and the coefficient of restitution  $\epsilon$ , which are physical parameters and can be obtained through experiments. The frictional behavior between the girder and the pier and between the girder and the glass wall is considered differently, with values of 0.05 and 0.0, respectively. The coefficient of restitution is considered to be zero for resting contact and



**Fig. 13.** Bridge washout: comparison of bridge-girder translational motion in (left) horizontal and (right) vertical axis between the averaged experimental results with adjustment and the simulation results performed by our ISPH-DEM code for initial height of: (a) 250 mm, (b) 300 mm and (c) 350 mm. The results performed by [46] by using a penalty-based method are presented in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

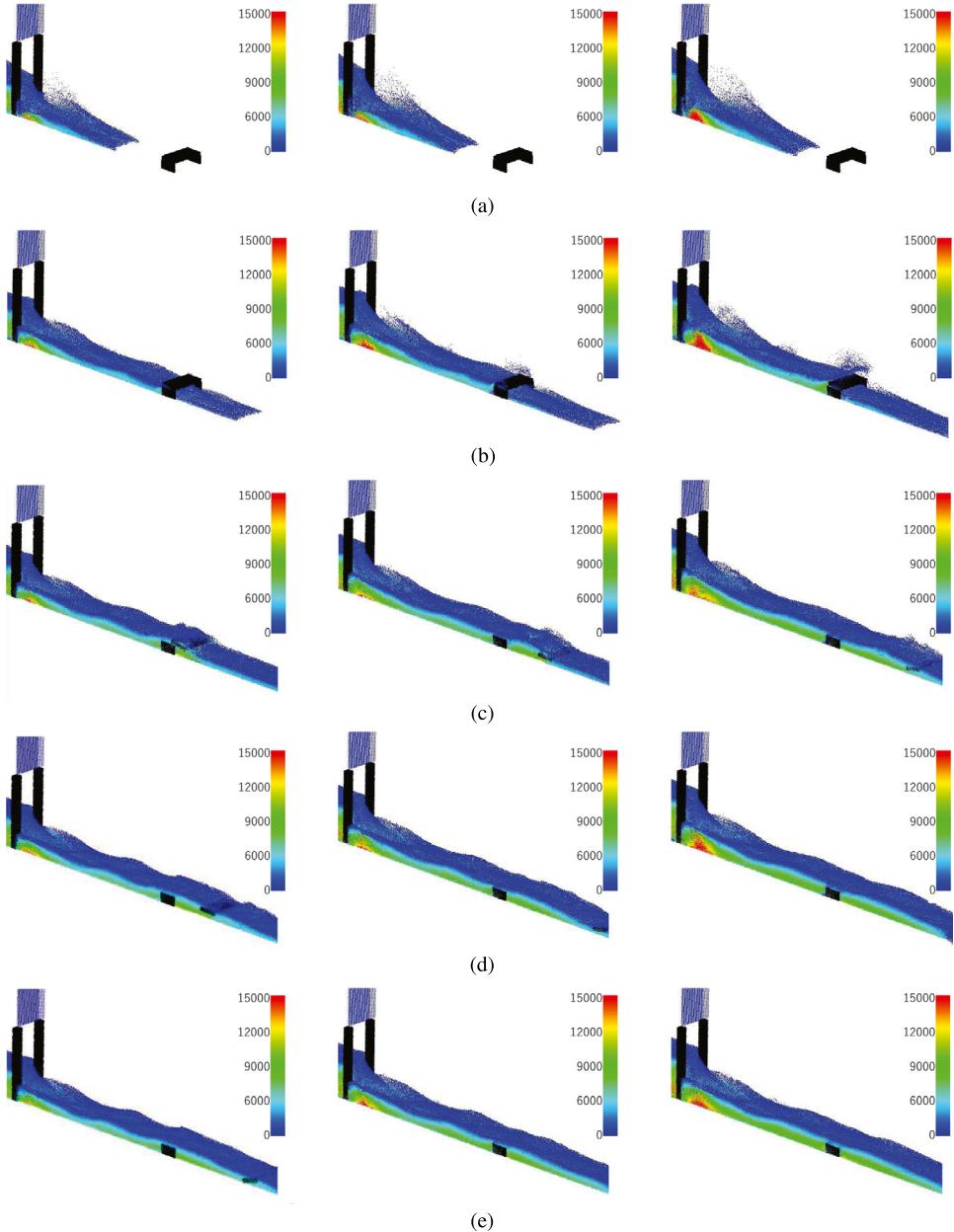
non-zero for collision,  $\epsilon = 0.8$  following [46,52]. Furthermore, the speed of opening gate is considered as free-fall velocity, similar to the load-pulley-gate opening mechanism which was used in the experimental tests.

The motion of the girder upon washout obtained from the experiments and simulations is compared within an observation area of 1 m around the initial position of the girder. Fig. 13 shows the average position of girder's center tracked during the washout in the x and y-axis referred from the world frame denoted as point **O** in Fig. 12.



**Fig. 14.** Bridge washout: comparison of bridge-girder rotational angle within  $z$ -axis  $\theta_z$  between the experimental results and the simulation results performed by our ISPH-DEM code for initial height of: (a) 250 mm, (b) 300 mm and (c) 350 mm. The results performed by [46] by using a penalty-based method are presented in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

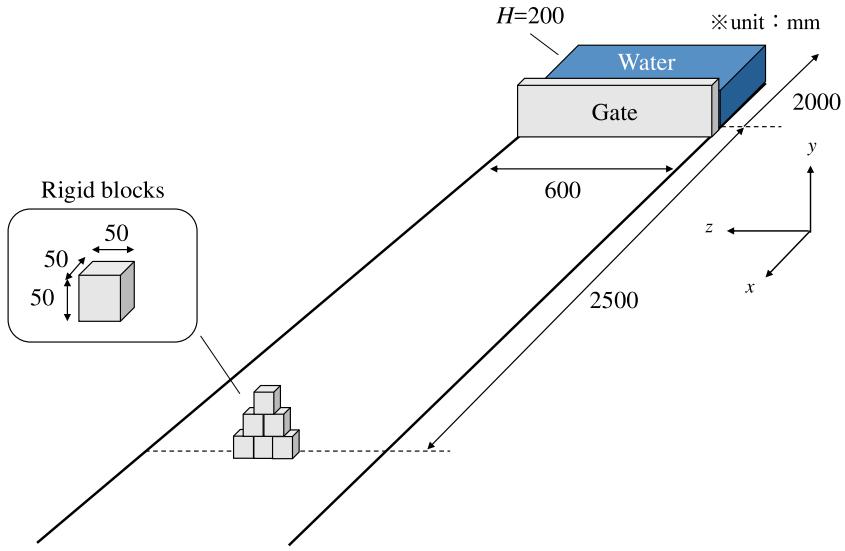
Furthermore, Fig. 14 shows the comparison of girder's rotational angle  $\theta_z$  within the  $z$ -axis. In Fig. 14, the positive angle indicates a clockwise rotation, while the negative angle represents the counter-clockwise rotation. From the two aforementioned figures, the performance and accuracy of the implemented ISPH-DEM with ETI contact are



**Fig. 15.** Bridge washout: time sequence of the 3D box girder bridge washout simulations for initial water height; (left) 250 mm, (center) 300 mm, and (right) 350 mm at time: (a)  $t = 0.5$  s, (b)  $t = 1.0$  s, (c)  $t = 1.5$  s, (d)  $t = 1.75$  s, and (e)  $t = 2.0$  s. Color contour plots pressure measured in  $\text{g cm}^{-1} \text{s}^{-2} \equiv 0.1 \text{ Pa}$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

validated for single rigid body with different particle resolutions. The time evolution of the washout simulations for  $d = 0.5 \text{ cm}$  are presented in Fig. 15.

The results presented by [46,52] considering penalty methods with  $d = 0.5 \text{ cm}$  were plotted for comparison. Here, [46,52] assumed a nonlinear Hertz–Mindlin penalty contact [53] with Poisson’s ratio and Young’s modulus of  $\nu = 0.28$  and  $E = 5 \text{ GPa}$  to compute the penalty spring and dashpot parameters. As can be noticed from Fig. 13, even though the penalty approach (colored in red) shows a reasonably accurate result on the rigid-body horizontal motion, it produces a remarkably less accurate result in comparison to the ETI contact algorithm, particularly for



**Fig. 16.** Fluid–rigid-block interaction: schematic illustration of the first configuration with small blocks stacked in a triangular formation (image not to scale).

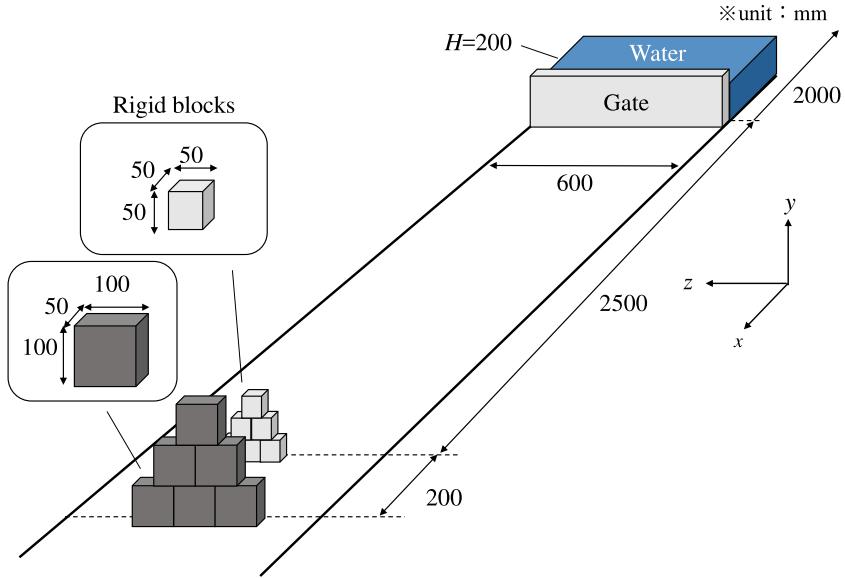
simulation of resting contact. A noticeable oscillation is observed at the beginning of the simulation for all cases (see Fig. 13 (right)). As pointed by [31], the implemented ETI contact algorithm is able to handle accurate treatments of resting contact and collision for both single- and multiple-point contacts, such as point-to-surface, edge-to-surface, or surface-to-surface contact. It should be noted that the ETI method presented does not require any calibration of non-physical contact parameters, something which is necessary for penalty-based methods.

#### 4.3. Rigid blocks stacked in a triangular formation and a dam break

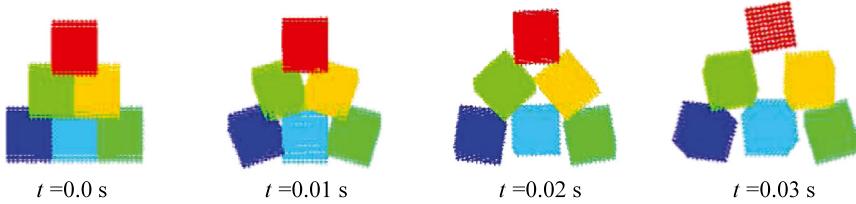
The next validation tests were conducted to validate the proposed formulation for fluid–multiple-rigid-body interaction. Two configurations of experiments were conducted in the same wave channel as in Section 4.1, though arranged in a different setting. The first configuration involves 6 “small” rigid blocks attacked by a dam-break (see Fig. 16). Related to this work, the blocks will be denoted from hereon either as “small” or “large”, each with a dimension of 50 mm × 50 mm × 50 mm or 100 mm × 50 mm × 100 mm, respectively. In the second configuration, both the small and large blocks are employed, yielding 12 blocks in total, arranged in two layers as in Fig. 17. In the following simulations, the friction contact between blocks should be resolved as accurately as possible both in its initial stacking formation and during the washout.

Previously, Canelas et al. [19] conducted a similar experimental test and performed numerical simulations using the SPH–DCDEM method. Furthermore, Junior et al. [47] reproduced the same experimental test as [19] with an MPS–DEM coupled approach, where they utilized a sub-cycling algorithm to avoid the limitation on time-step size. As discussed previously, the penalty-family approach generally requires a significantly smaller time-step size to achieve stability during resting and surface-to-surface contact. In addition to that, a calibration of artificial penalty parameters, e.g. the spring modulus  $k$  and damping coefficient  $c$ , is generally required as the accuracy of the approximated repulsive forces is sensitive to those parameters. These penalty parameters are influenced by various factors, such as, among others, the particle resolution, the cumulative rigid body mass, as well as by the prescribed Neumann boundary condition and other external loadings. They are, therefore, problem-specific and may not be used straightforward for different simulations with different conditions.

For instance, when the penalty parameters are set to be too “soft”, penetration between surfaces might occur as the approximated penalty force unable to fully satisfy the contact constraint conditions, Eqs. (46) and (47). However, on contrary, when these parameters are set to be too large (or too “stiff”), the contact force can be overestimated,



**Fig. 17.** Fluid–rigid-block interaction: schematic illustration of the second configuration with small and large blocks stacked in two-layer triangular formation (image not to scale).



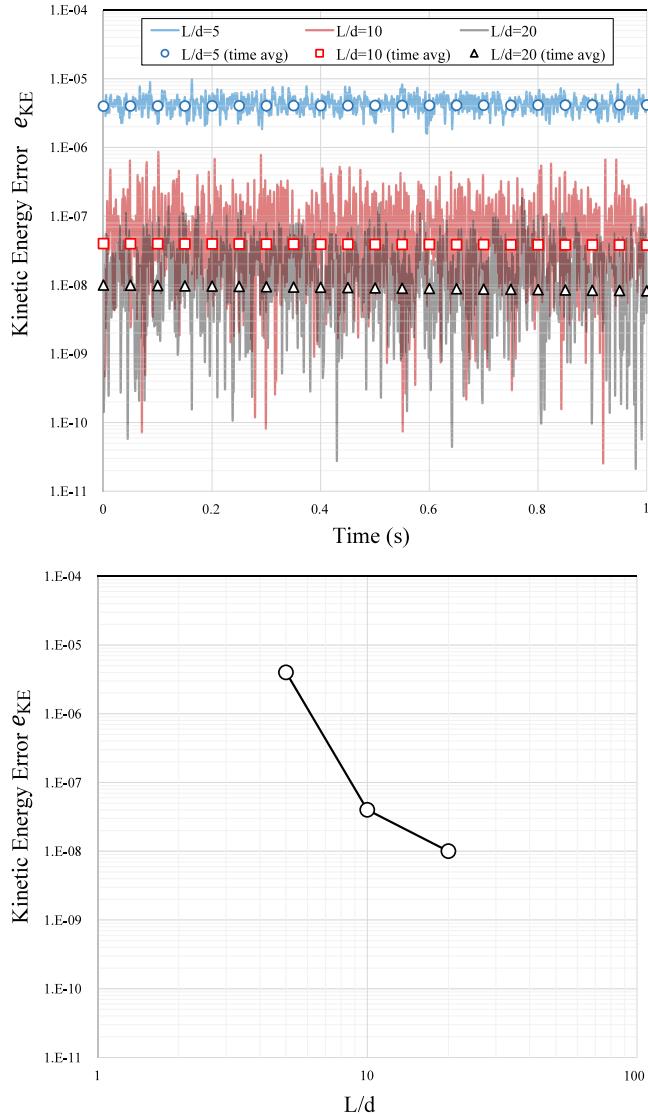
**Fig. 18.** Fluid–rigid-block interaction: numerical instability of the penalty method causing the triangular formation to explode if  $\Delta t = 0.001$  s is assumed.

and thus, results in numerical instability if the chosen time-step size is larger than a certain critical threshold. Fig. 18 shows a simulation result of unstable triangular formation when the nonlinear Hertz–Mindlin penalty contact method [53] is employed within the same range of time-step size as the fluid solver, i.e.  $\Delta t = 0.001$  s. The assumed Poisson’s ratio and Young’s modulus to compute the penalty spring and dashpot parameters are  $\nu = 0.28$  and  $E = 20$  GPa. The coefficient of restitution between all blocks and wall boundary is set as 0.35, whereas the mass density of the blocks is set as  $960 \text{ kg/m}^3$ . With the aforementioned penalty parameters, the critical time-step size should be in the order of  $\Delta t \sim 10^{-5}$  s following a corollary of Rayleigh’s theorem [54]:

$$\Delta t \leq \Delta t_{\text{crit}}^{\text{penalty}} = 2\sqrt{\frac{M}{k}} \left( \sqrt{\xi^2 + 1} - \xi \right), \quad (66)$$

where,  $M$  is taken as the mass of the rigid body, composed of several particles, and  $\xi = \frac{c}{2\sqrt{Mk}}$  is a damping ratio which depends on the assumed spring modulus  $k$  and damping coefficient  $c$ .

The ETI, on the other hand, does not require any calibration of artificial contact parameters, and it also exhibits less dependency on temporal and spatial discretization parameters. As has been previously discussed in [31], the errors generated from different time-step sizes stem from the lack of contact detection algorithm in our simulations, and thus, overshooting of velocities at the time of impact might occur due to the numerical integration, even before the ETI solver performs the post-collision velocity computations. Our study shows that time increments smaller than 0.001 s yield almost the same solutions for all of our examples. This is a noteworthy advantage of ETI, in



**Fig. 19.** Fluid–rigid-block interaction: convergence analysis for different particle resolutions: (up)  $e_{KE}$  vs time; (bottom) time-averaged  $e_{KE}$  vs  $L/d$ , where  $L = 5$  cm denotes the side length of the blocks and  $d$  is the particle diameter.  $\Delta t = 0.001$  s is assumed for all cases.

particular, because it lies within the same range as the time step used by the ISPH solver. Furthermore, as the ETI contact is evaluated on the exact rigid-body surface points [31] (see Eq. (35)), the dependency of particle resolutions should be vanishingly small provided that the intricate feature of the rigid-body surface is adequately represented. To quantify the convergence of error caused by particle refinement, Fig. 19 was developed by measuring the total kinetic energy,  $\hat{E}_k$ , of a triangular formation comprising of 6 small blocks at rest, which can be calculated as:

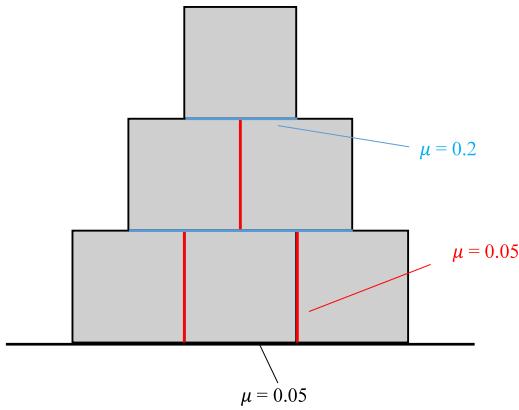
$$\hat{E}_k = \hat{E}_t + \hat{E}_r = \sum_{RB=1}^6 \left( \frac{1}{2} M_{RB} \mathbf{V}_{RB} \cdot \mathbf{V}_{RB} + \frac{1}{2} \boldsymbol{\omega}_{RB}^T \mathbf{I}_{RB} \boldsymbol{\omega}_{RB} \right). \quad (67)$$

Here,  $\hat{E}_t$  and  $\hat{E}_r$  are the total translational and the rotational kinetic energy of the formation, respectively. To avoid dependency on dimension, a normalized kinetic energy error  $e_{KE}^{n+1} = \hat{E}_k^{n+1}/\hat{E}_k^*$  is presented by normalizing the total kinetic energy after the ETI calculation with the intermediate value considering external forces. Note that, ideally,

**Table 3**

Fluid–rigid-block interaction: simulation parameters considered.

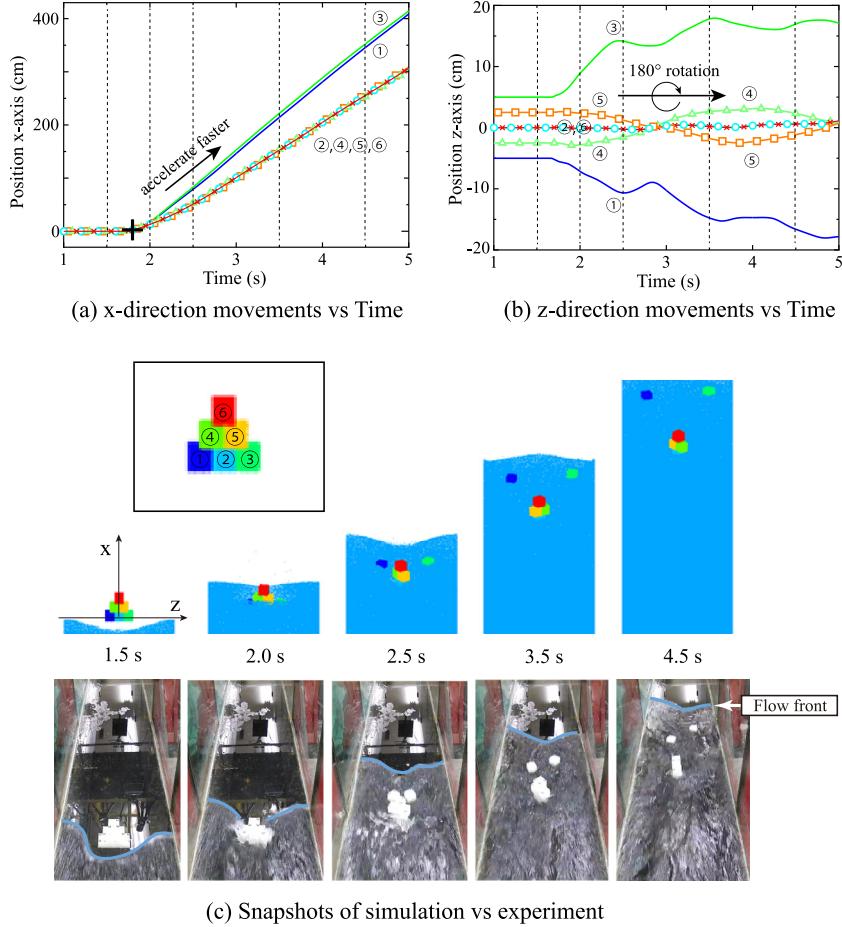
|                                      |   |
|--------------------------------------|---|
| Particle diameter ( $d$ )            | 0.5 cm  |
| Time step ( $\Delta t$ )             | 0.001 s   |
| Fluid:                               |   |
| Density ( $\rho_w$ )                 | 1000 kg/m <sup>3</sup>                            |
| Dynamic viscosity ( $\mu_w$ )        | $8.9 \times 10^{-4}$ Pa·s                         |
| Relaxation coefficient ( $\alpha$ )  | 0.01  |
| Smagorinsky constant ( $C_s$ )       | 0.1   |
| Ratio of slip and non-slip condition | 2:1   |
| Rigid blocks:                        |   |
| Density ( $\rho$ )                   | 960 kg/m <sup>3</sup>                             |
| Friction coeff. ( $\mu$ )            | see Fig. 20 (same size);<br>0.07 (different size) |
| Coeff. restitution ( $\epsilon$ )    | 0.0 (rest); 0.35 (collision)                      |

**Fig. 20.** Fluid–rigid-block interaction: coefficient of friction settings for a single triangular stack.

these values should be equal to zero to denote a fully resting contact. In the upcoming validation work,  $d = 0.5$  cm is considered for all simulations, resulting in a total of about 4.1 million particles.

An open issue to be addressed is related to the dynamic friction law for wet surfaces. Previous works that attempted to accurately represent the surface-to-surface frictional contact under a wet condition [19,47] showed that the conventional Coulomb-type dynamic friction law often overestimates the frictional behavior in wet conditions. Related to this issue, more in-depth studies have been conducted in both engineering [55,56] and physics [57]. For instance, Arakawa [55] proposed a dynamic friction model which depends on the interface fluid thickness and viscosity, as well as the sliding velocity, where Zhu et al. [56] developed a friction coefficient equation to model hydroplaning phenomena on a car tire. In the current work, the dynamic friction contact under wet condition is simply modeled by reducing the dynamic friction coefficient to be about 25% of the dynamic friction coefficient in dry condition. This reduction of friction coefficient is applied at the bottom layer of the stack and the lateral surfaces between blocks (see Fig. 20). There, the coefficient of friction is set as  $\mu = 0.05$ , whereas at the other surfaces  $\mu = 0.2$ . For the second configuration, the coefficient of friction between the small and the large block is set as  $\mu = 0.07$ , considering that the contact between blocks with different sizes only occurs during wet conditions. All the numerical parameters used in the simulations are listed in Table 3.

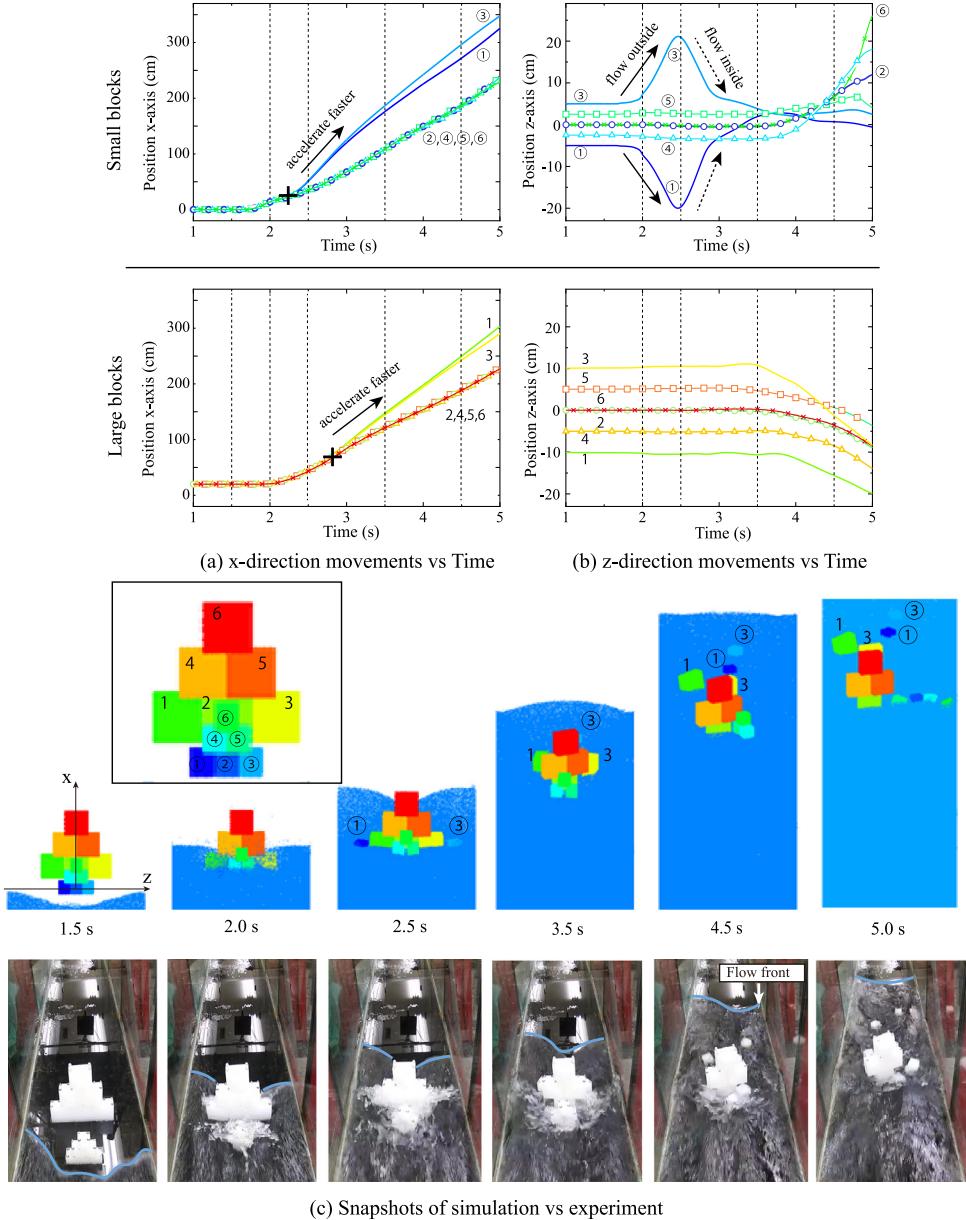
For the first configuration, the movements of the six small blocks in the x and z-axis are plotted in Fig. 21(a) and (b). On top of that, the visual comparison between the conducted simulation and experiment is presented in



**Fig. 21.** Fluid–rigid-block interaction: simulation results vs experimental data of the first configuration with six small blocks stacked in a triangular formation.

**Fig. 21(c)** as a series of snapshots taken at  $t = 1.5, 2.0, 2.5, 3.5$ , and  $4.5$  s after the gate is opened. At around  $t = 1.5$  s, the waterfront hits the blocks and begins to transport them downstream. The bottom left and right blocks, denoted as block ① and ③ in **Fig. 21**, are washed away first towards the left and right sides, away from the centerline, respectively. The other four blocks, ②, ④, ⑤, and ⑥, stay in the stacking “cross” formation and move together towards the downstream with an observed rotational motion. These behaviors are also observed in the conducted experiments as shown in **Fig. 21(c)**. The movements of each block in our study are different from the previous works done by Canelas et al. [19]. In their tests with similar triangular formation, there is a 5-cm gap separating the blocks within the same stacking level, which is about 1/3 of their block size. In their study, the stacking formation is destroyed right after the dam-break hits the formation, causing the blocks to washout separately. It is worth highlighting the capability of the ETI method to accurately simulate the stacking cross formation of the four center blocks and their movement as a clump towards the downstream. As explained by [31], proper contact identification and priority sorting are important to achieve a stable and robust impulse propagation for multiple rigid-body problems.

In the second configuration, a dam-break interacting with 12 blocks arranged in two triangular formation is adopted. The movement of the six small blocks and six large blocks in x and z-axis is plotted in **Fig. 22(a)** and (b), respectively. Moreover, the snapshots given in **Fig. 22(c)** show the comparison of the obtained numerical results and the experimental data at  $t = 1.5, 2.0, 2.5, 3.5, 4.5$ , and  $5.0$  s. Similar to the first configuration, when the water front hits the small triangular stack, the blocks begin to get washed away downstream. Here, the bottom left and right



**Fig. 22.** Fluid–rigid-block interaction: simulation results vs experimental data of the second configuration with twelve blocks stacked in two layers of triangular formation.

blocks are also washed away towards the sides, flowing faster than the remaining four blocks and passing through the sides of the large triangular stack. The remaining four small blocks similarly stay in a cross formation, which move and hit the larger triangular stack together with the water flow. The outermost left and right blocks of the large triangular stack begin to flow faster than other four blocks after  $t = 3.5$  s. The small blocks remain moving in their cross formation until around  $t = 4.5$  s, where they finally collapse and flow separately. The produced numerical simulations are able to reproduce the stacking mechanism of multiple rigid bodies, along with frictional transport with dynamic friction and blocks collision. All of these trends are similar to the data obtained from the conducted experiments. It is important to note, however, that the experimental and computational results still present some small discrepancies, and further investigations, particularly on the wet frictional contact and collision, are necessary.

**Table 4**

Fluid–bunny interaction: simulation parameters considered.

|                                      |                                     |
|--------------------------------------|-------------------------------------|
| Particle diameter ( $d$ )            | 1.0 cm                              |
| Time step ( $\Delta t$ )             | 0.0005 s                            |
| Fluid:                               |                                     |
| Density ( $\rho_w$ )                 | 1000 kg/m <sup>3</sup>              |
| Dynamic viscosity ( $\mu_w$ )        | 8.9 × 10 <sup>-4</sup> Pa · s       |
| Relaxation coefficient ( $\alpha$ )  | 0.01                                |
| Smagorinsky constant ( $C_s$ )       | 0.2                                 |
| Ratio of slip and non-slip condition | 2:1                                 |
| Bunnies:                             |                                     |
| Density ( $\rho$ )                   | 300 kg/m <sup>3</sup>               |
| Friction coeff. ( $\mu$ )            | 0.2 (bunny–bunny); 0.5 (bunny–bowl) |
| Coeff. restitution ( $\epsilon$ )    | 0.5                                 |

To conclude, the current validation works showcase the strength of the proposed ETI method for fluid–multiple-rigid-body interaction. The ETI contact method is able to represent accurate stacking mechanism, frictional sliding behavior, as well as stable collision between blocks with various sizes. All in all, the ETI method allows a significantly larger time increment than the critical step allowed by the penalty method. This advantage supports the usage of conventional staggered two-way SPH–DEM coupling scheme more efficiently.

#### 4.4. Fluid interacting with objects with complicated shape

In the current subsection, the robustness of our fluid–rigid-body coupling scheme is tested for objects with a complicated shape. Here, the Stanford bunny model was considered and subjected to a free-surface fluid flow. The model size of the current numerical test is depicted in top and side view by Fig. 23. The size of each bunny is 51 cm × 53 cm × 38 cm, with density and coefficient of restitution of 300 kg/m<sup>3</sup> and 0.5, respectively. The bunny is discretized as a cluster of particles and  $d = 1$  cm is chosen as the particle diameter, and hence, the total number of particles altogether is about 5.8 million. The time increment for this simulation is fixed to be 0.0005 s. The coefficient of friction is set to be 0.2 among the bunnies and 0.5 between each bunny and the half-sphere bowl. The considered numerical parameters for the simulations are summarized in Table 4.

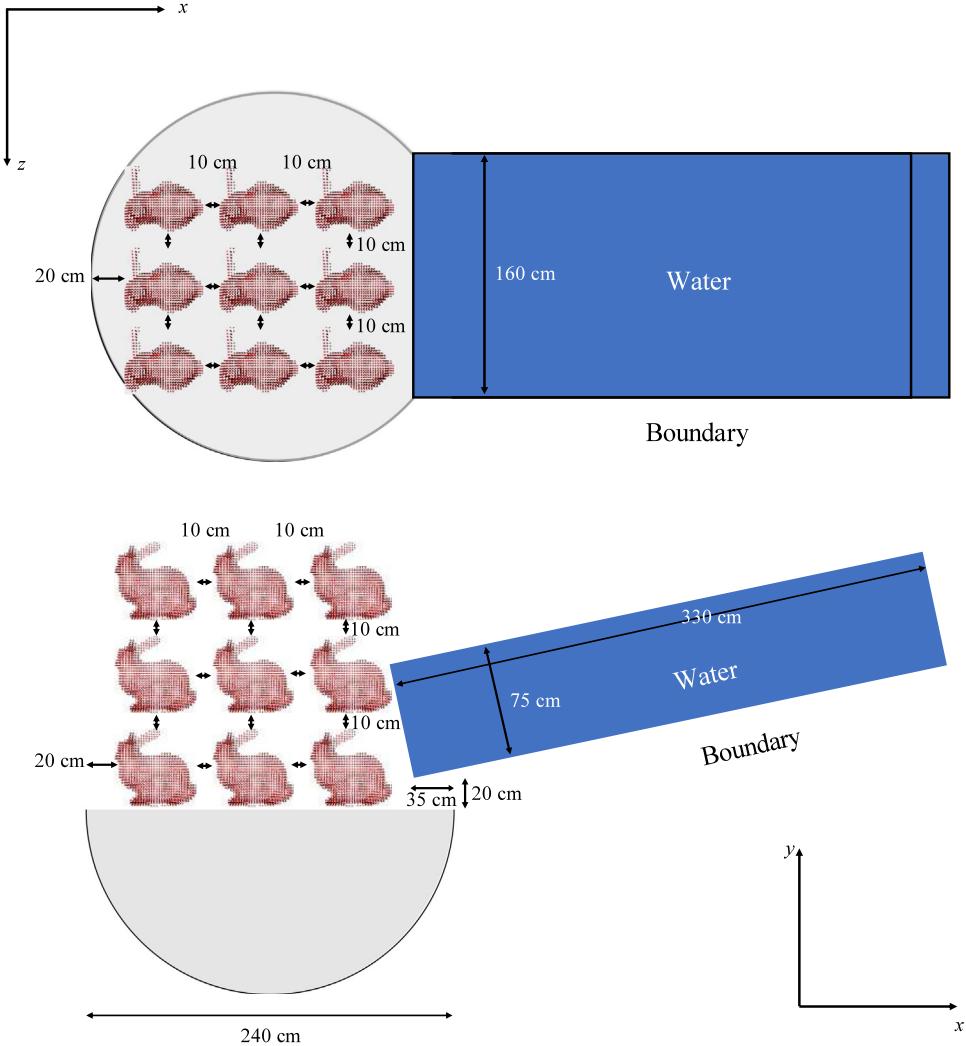
The simulations include two sequential steps: all the bunnies fall to the bottom of the bowl, and then, water was poured into the bowl where some of the bunnies were washed out of the bowl. The simulation results, which demonstrate the stability and robustness of the implementation to handle contact of rigid bodies with complicated shape, are shown in Fig. 24.

## 5. Conclusions

In the current work, we focused on the development of a particle-based fluid–structure interaction (FSI) simulator using an ISPH–DEM coupled framework for simulating violent free-surface flows interacting with multiple rigid bodies. The ETI method, which was originally proposed by Tang and colleagues [30] and generalized for particle-discretized rigid bodies by [31], is adopted for the particle-based FSI simulations to refine the collisions response between rigid surfaces. Meanwhile, the ISPH method was implemented to produce a smooth and accurate pressure distribution of free-surface fluid flow with breaking and fragmentation.

The main difficulties in simulating particle-based FSI simulations appear in the accuracy of collision and contact response for rigid-body dynamic simulations, which includes resting and collision contacts of multiple points and surfaces. Within the framework of DEM, compared to the commonly-used penalty-based family approach, the ETI method allows a fixed moderate time increment, yielding a more robust and accurate contact modeling within the same order of time-step size as the one used in the ISPH fluid simulation. The conventional two-way, staggered, weakly coupled fluid–structure coupling scheme can be utilized efficiently with help of the ETI method.

The current work also covers three experimental validation tests, which were conducted to assure the quality and robustness of the coupled ISPH–DEM implementation. Two experiments were designed and conducted to



**Fig. 23.** Fluid–bunny interaction: details of the fluid and bunnies initial condition.

validate the accuracy and robustness of the fluid–rigid-body coupling scheme, for both single and multiple objects. The numerical results obtained are in very good agreement with the obtained experimental test. Despite this, the authors believe that there is always room for improvement, particularly in tailoring the ETI approach to handle dynamics-friction contact models in wet conditions, i.e. when a thin layer of fluid presents between the contacting surface. Finally, a demonstration simulation of complex-shaped rigid bodies, which are subjected to a large number of constraints, was conducted to show the performance of the proposed method in tackling different numerical instabilities that occur during collision and resting contacts of arbitrary-shaped objects.

As for future works, further research should focus on improving the efficiency of the ETI solver, considering parallelism employing contact island algorithm. It is also interesting to investigate the factors that may improve the accuracy and stability of the coupling scheme. For instance, when light structures are involved, a strongly coupled FSI approach will be a promising option, and may result in a more stable coupling condition as the kinematic and traction equilibriums at the interface are enforced iteratively through a converging feedback loop. Extension of the ETI method to handle deformable bodies is also an undergoing research topic.



**Fig. 24.** Fluid–bunny interaction: visualized simulation results (left) along with and (right) without water.

## CRediT authorship contribution statement

**Mitsuteru Asai:** Conceptualization, Methodology, Supervision, Funding acquisition, Writing - review & editing. **Yi Li:** Conceptualization, Methodology, Software, Investigation, Validation, Formal analysis. **Bodhinanda Chandra:** Conceptualization, Methodology, Software, Formal analysis, Visualization, Writing - original draft, Review & editing. **Shinsuke Takase:** Investigation, Validation, Resources.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

This work was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Nos. JP-20H02418, 19H01098, and 19H00812. The authors also received computational environment support through the “Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures” in Japan (Project ID: jh200034-NAH and jh200015-NAH). The authors would like to thank Prof. K. Izuno from Ritsumeikan University and Prof. D. Isobe from the University of Tsukuba for renting them their experimental devices including the water channel and the 3D motion capture system. The authors gratefully acknowledge the help from Prof. M. Isshiki from Ehime University for creating the visualization graphics for the last numerical examples.

## Data availability statement

The data that support the findings of this study are available from the corresponding authors upon reasonable request.

## Appendix A. Unit quaternions

To update a rigid body orientation, the unit quaternion is often used and preferred as it overcomes the Gimbal lock [48] and the singularity problems [49] arise from the Euler angles and rotation matrices approach, respectively [50,51]. Introduced by Sir Rowan Hamilton [58], a quaternion is defined as,

$$\mathbf{q} = [q_0; \mathbf{q}_V] = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}, \quad (\text{A.1})$$

where  $q_0, q_1, q_2$  and  $q_3$  are real numbers which is called as the *constituents*, and  $\mathbf{i}, \mathbf{j}$  and  $\mathbf{k}$  are the unit vectors directed along to the x, y and z-axis, respectively. The relationship among these unit vectors follows the system of imaginary equations defined as,

$$\begin{aligned} \mathbf{i}^2 &= \mathbf{j}^2 = \mathbf{k}^2 = -1; \\ \mathbf{ij} &= \mathbf{k}, \mathbf{jk} = \mathbf{i}, \mathbf{ki} = \mathbf{j}; \\ \mathbf{ji} &= -\mathbf{k}, \mathbf{kj} = -\mathbf{i}, \mathbf{ik} = -\mathbf{j}, \end{aligned} \quad (\text{A.2})$$

where no linear relation between  $\mathbf{i}, \mathbf{j}, \mathbf{k}$  being supposed to exist.

Three useful operations of a quaternion: the conjugation, norm, and inverse are expressed respectively by Eqs. (A.3)~(A.5):

$$\bar{\mathbf{q}} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}, \quad (\text{A.3})$$

$$\|\mathbf{q}\| = \sqrt{\mathbf{q}\bar{\mathbf{q}}} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}, \quad (\text{A.4})$$

$$\mathbf{q}^{-1} = \frac{\bar{\mathbf{q}}}{\|\mathbf{q}\|}. \quad (\text{A.5})$$

A unit quaternion is defined as a quaternion whose norm is 1, which by extent, having its inverse equal to its own conjugate,

$$\|\mathbf{q}\| = \sqrt{\mathbf{q}\bar{\mathbf{q}}} = 1 \longrightarrow \mathbf{q}^{-1} = \bar{\mathbf{q}}. \quad (\text{A.6})$$

Quaternions are added or subtracted by adding or subtracting their constituents, so that,

$$\mathbf{p} \pm \mathbf{q} = (p_0 \pm q_0) + (p_1 \pm q_1)\mathbf{i} + (p_2 \pm q_2)\mathbf{j} + (p_3 \pm q_3)\mathbf{k}. \quad (\text{A.7})$$

In virtue of the definitions written in Eq. (A.2), their multiplication is effected by the following vector formulation,

$$\begin{aligned} \mathbf{p} \wedge \mathbf{q} &= (p_0 + p_1\mathbf{i} + p_2\mathbf{j} + p_3\mathbf{k}) \wedge (q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}) \\ &= (p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3) + \\ &\quad (p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2)\mathbf{i} + \\ &\quad (p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1)\mathbf{j} + \\ &\quad (p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0)\mathbf{k}, \end{aligned} \quad (\text{A.8})$$

or in vector representation can be expressed as,

$$\mathbf{p} \wedge \mathbf{q} = [p_0q_0 - \mathbf{p}_V \cdot \mathbf{q}_V; p_0\mathbf{q}_V + q_0\mathbf{p}_V + \mathbf{p}_V \times \mathbf{q}_V]. \quad (\text{A.9})$$

The multiplication between two quaternions is denoted here by “ $\wedge$ ” operator, which consists of vector dot and cross products, and thus, inherits a non-commutative property.

In many research papers, e.g. [59,60], the integration of unit quaternions is often solved employing Taylor's series, where a time derivative increment is added to the original unit quaternion  $\mathbf{q}^n$ . This algorithm usually adds an increment which is not exactly of zero length, and therefore, leading to the length of  $\mathbf{q}^{n+1}$  will be different from unity. Kleppmann [61] proved clearly that if the body in question is rotating, a standard numerical ODE solver

cannot preserve a quaternion's property of unit magnitude. This integration error is, furthermore, known as the quaternion drift in the literature.

In the current work, a direct multiplication approach developed by [62] is used to substantially preserve the unit quaternions' length. As presented in [63,64], this approach can be derived by introducing an exponential mapping and performing a Maclaurin series into the ODE. One can furthermore write the variation of unit quaternions  $\mathbf{dq}^{n+1}$  as,

$$\mathbf{dq}^{n+1} = \left[ \cos\left(\frac{\|\boldsymbol{\omega}^{n+1}\| \Delta t}{2}\right); \frac{\boldsymbol{\omega}^{n+1}}{\|\boldsymbol{\omega}^{n+1}\|} \sin\left(\frac{\|\boldsymbol{\omega}^{n+1}\| \Delta t}{2}\right) \right], \quad (\text{A.10})$$

$$\text{where, } \|\boldsymbol{\omega}\| = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}. \quad (\text{A.11})$$

Following this step, the new unit quaternions can be computed by taking a quaternion multiplication between the variation  $\mathbf{dq}^{n+1}$  and the current unit quaternion  $\mathbf{q}^n$  as,

$$\mathbf{q}^{n+1} = \mathbf{dq}^{n+1} \wedge \mathbf{q}^n. \quad (\text{A.12})$$

Here, the initial unit quaternions at  $t = 0$ , denoted as  $\mathbf{q}^0$ , is set at the beginning of the simulation as,

$$\mathbf{q}^0 = [1, 0, 0, 0]. \quad (\text{A.13})$$

Even though the direct multiplication technique explained by Eqs. (A.10) and (A.12) can address the quaternion drift significantly, slight errors might occur during the integration process, which is often caused by an accumulation of numerical truncation. To preserve the unity constraint ultimately, a re-normalization can be applied to the updated unit quaternions from time to time as,

$$\hat{\mathbf{q}} = \frac{\mathbf{q}}{\|\mathbf{q}\|}. \quad (\text{A.14})$$

## Appendix B. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.cma.2021.113681>.

## References

- [1] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, *Astron. J.* 82 (1977) 1013–1024.
- [2] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: Theory and application to non-spherical stars, *Mon. Not. R. Astron. Soc.* 181 (3) (1977) 375–389.
- [3] J.J. Monaghan, Simulating free surface flows with SPH, *J. Comput. Phys.* 110 (2) (1994) 399–406.
- [4] S.J. Cummins, M. Rudman, An SPH projection method, *J. Comput. Phys.* 152 (2) (1999) 584–607.
- [5] S. Koshizuka, Y. Oka, Moving-particle semi-implicit method for fragmentation of incompressible fluid, *Nucl. Sci. Eng.* 123 (3) (1996) 421–434.
- [6] C. Ulrich, M. Leonardi, T. Rung, Multi-physics SPH simulation of complex marine-engineering hydrodynamic problems, *Ocean Eng.* 64 (2013) 109–121, <http://dx.doi.org/10.1016/j.oceaneng.2013.02.007>.
- [7] E. Harada, H. Gotoh, H. Ikari, A. Khayyer, Numerical simulation for sediment transport using MPS-DEM coupling model, *Adv. Water Resour.* 129 (2017) <http://dx.doi.org/10.1016/j.advwatres.2017.08.007>.
- [8] S. Shao, Incompressible SPH flow model for wave interactions with porous media, *Coast. Eng.* 57 (3) (2010) 304–316.
- [9] H. Akbari, M.M. Namin, Moving particle method for modeling wave interaction with porous structures, *Coast. Eng.* 74 (2013) 59–73.
- [10] A. Mahmoud Aly, M. Asai, Three-dimensional incompressible smoothed particle hydrodynamics for simulating fluid flows through porous structures, *Transp. Porous Media* 110 (2015) <http://dx.doi.org/10.1007/s11242-015-0568-8>.
- [11] X. Sun, M. Sakai, M.-T. Sakai, Y. Yamada, A Lagrangian-Lagrangian coupled method for three-dimensional solid–liquid flows involving free surfaces in a rotating cylindrical tank, *Chem. Eng. J.* 246 (2014) 122–141, <http://dx.doi.org/10.1016/j.cej.2014.02.049>.
- [12] D. Violeau, B.D. Rogers, Smoothed particle hydrodynamics (SPH) for free-surface flows: Past, present and future, *J. Hydraul. Res.* (2016) 1–26.
- [13] M.-B. Liu, G.-R. Liu, Particle Methods for Multi-Scale and Multi-Physics, World Scientific, 2015.
- [14] X. Liu, H. Xu, S. Shao, P. Lin, An improved incompressible SPH model for simulation of wave–structure interaction, *Comput. & Fluids* 71 (2013) 113–123.
- [15] T. Takahashi, C.M. Lin, A multilevel SPH solver with unified solid boundary handling, in: Pacific Graphics 2016, vol. 35, Wiley & Sons, 2016, pp. 207–220.
- [16] A. Rafiee, K.P. Thiagarajan, An SPH projection method for simulating fluid-hypoelastic structure interaction, *Comput. Methods Appl. Mech. Engrg.* 198 (33) (2009) 2785–2795.
- [17] A. Khayyer, H. Gotoh, H. Falahaty, Y. Shimizu, An enhanced ISPH-SPH coupled method for simulation of incompressible fluid–elastic structure interactions, *Comput. Phys. Comm.* 232 (2018) 139–164.

- [18] T. Harada, Real-time rigid body simulation on GPUs, in: GPU Gems, vol. 3, 2007, pp. 611–632.
- [19] R.B. Canelas, A.J. Crespo, J.M. Domínguez, R.M. Ferreira, M. Gómez-Gesteira, SPH-DCDEM model for arbitrary geometries in free surface solid-fluid flows, *Comput. Phys. Comm.* 202 (2016) 131–140.
- [20] J.C. Joubert, D.N. Wilke, N. Govender, P. Pizette, U. Tuzun, N.-E. Adriak, 3D gradient corrected SPH for fully resolved particle-fluid interactions, *Appl. Math. Model.* (2020) 816–840.
- [21] P.A. Cundall, O.D. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1) (1979) 47–65.
- [22] G.-H. Shi, Discontinuous deformation analysis: A new numerical model for the statics and dynamics of deformable block structures, *Eng. Comput.* 9 (2) (1992) 157–168.
- [23] N. Kikuchi, J.T. Oden, Contact Problems in Elasticity, Society for Industrial and Applied Mathematics, 1988, <http://dx.doi.org/10.1137/1.978161970845>, arXiv:<https://pubs.siam.org/doi/pdf/10.1137/1.978161970845>, URL <https://pubs.siam.org/doi/abs/10.1137/1.978161970845>.
- [24] R. Barzel, A.H. Barr, A modeling system based on dynamic constraints, in: ACM SIGGRAPH Computer Graphics, Vol. 22, ACM, 1988, pp. 179–188.
- [25] D. Baraff, Analytical methods for dynamic simulation of non-penetrating rigid bodies, in: ACM SIGGRAPH Computer Graphics, Vol. 23–3, ACM, 1989, pp. 223–232.
- [26] D.E. Stewart, Rigid-body dynamics with friction and impact, *SIAM Rev.* 42 (1) (2000) 3–39.
- [27] J.K. Hahn, Realistic animation of rigid bodies, in: ACM SIGGRAPH Computer Graphics, Vol. 22–4, ACM, 1988, pp. 299–308.
- [28] B.V. Mirtich, Impulse-Based Dynamic Simulation of Rigid Body Systems, University of California, Berkeley, 1996.
- [29] W.J. Stronge, Impact Mechanics, Cambridge university press, 2018.
- [30] X. Tang, A. Paluszny, R.W. Zimmerman, An impulse-based energy tracking method for collision resolution, *Comput. Methods Appl. Mech. Engrg.* 278 (2014) 160–185.
- [31] Y. Li, M. Asai, B. Chandra, M. Ishiki, Energy-tracking impulse method for particle-discretized rigid-body simulations with frictional contact, *Comput. Part. Mech.* (2020) 1–22.
- [32] E.-S. Lee, C. Moulinec, R. Xu, D. Violeau, D. Laurence, P. Stansby, Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method, *J. Comput. Phys.* 227 (18) (2008) 8417–8436.
- [33] E.J. Parteli, Dem simulation of particles of complex shapes using the multisphere method: Application for additive manufacturing, in: AIP Conference Proceedings, Vol. 1542, AIP, 2013, pp. 185–188.
- [34] O. Jou Devesa, Theoretical Study About Sea Ice Behaviour, Analysis of Floe-Ice Fractures and Discrete Element Modelling for Ship-Ice Interactions, Universitat Politècnica de Catalunya, 2018.
- [35] A.J. Chorin, Numerical solution of the Navier-Stokes equations, *Math. Comp.* 22 (104) (1968) 745–762.
- [36] M. Asai, A.M. Aly, Y. Sonoda, Y. Sakai, A stabilized incompressible SPH method by relaxing the density invariance condition, *J. Appl. Math.* (2012) <http://dx.doi.org/10.1155/2012/139583>.
- [37] J.J. Monaghan, J.C. Lattanzio, A refined particle method for astrophysical problems, *Astron. Astrophys.* 149 (1985) 135–143.
- [38] J.J. Monaghan, Extrapolating B splines for interpolation, *J. Comput. Phys.* 60 (2) (1985) 253–262.
- [39] R. Xu, P. Stansby, D. Laurence, Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach, *J. Comput. Phys.* 228 (18) (2009) 6703–6725, <http://dx.doi.org/10.1016/j.jcp.2009.05.032>.
- [40] H. Gotoh, S. Shao, T. Memita, SPH-LES model for numerical investigation of wave interaction with partially immersed breakwater, *Coastal Eng. J.* 46 (01) (2004) 39–63.
- [41] S. Shao, H. Gotoh, Turbulence particle models for tracking free surfaces, *J. Hydraul. Res.* 43 (3) (2005) 276–289.
- [42] D. Violeau, R. Issa, Numerical modelling of complex turbulent free-surface flows with the SPH method: An overview, *Internat. J. Numer. Methods Fluids* 53 (2) (2007) 277–304.
- [43] G.-R. Liu, M.B. Liu, Smoothed Particle Hydrodynamics: A Meshfree Particle Method, World Scientific, 2003.
- [44] M. Asai, K. Fujimoto, S. Tanabe, M. Beppu, Slip and no-slip boundary treatment for particle simulation model with incompatible step-shaped boundaries by using a virtual maker, *Trans. Jpn. Soc. Comput. Eng. Sci.* 2013 (2013).
- [45] S. Koshizuka, A. Nobe, Y. Oka, Numerical analysis of breaking waves using the moving particle semi-implicit method, *Internat. J. Numer. Methods Fluids* 26 (1998) 751–769.
- [46] B. Chandra, M. Asai, Verification and validation of the fluid-rigid body interaction simulation by the smoothed particle hydrodynamics method, in: Proceedings of Computational Engineering Conference JSSES, vol. 21, 2016.
- [47] R.A.A. Junior, L.-Y. Cheng, P.H.S. Oselle, An improvement of rigid bodies contact for particle-based non-smooth walls modeling, *Comput. Part. Mech.* (2019) 561–580.
- [48] K. Shoemake, Animating rotation with quaternion curves, in: ACM SIGGRAPH Computer Graphics, Vol. 19–3, ACM, 1985, pp. 245–254.
- [49] B.M. Saunders, Fast Animation Dynamics (Ph.D. thesis), University of Cambridge, 2000.
- [50] D. Baraff, Physically based modeling: Rigid body simulation, in: SIGGRAPH Course Notes, ACM SIGGRAPH, vol. 2, (1) 2001.
- [51] D. Eberly, Quaternion Algebra and Calculus, Magic Software Inc, 2002.
- [52] M. Asai, B. Chandra, Numerical prediction of bridge wash-out during natural disaster by using a stabilized ISPH method, in: The 2016 World Congress on Advances in Civil, Environmental, and Materials Research, ACEM16, Jeju Island, Korea, 2016.
- [53] C. O’Sullivan, Particulate Discrete Element Modelling: A Geomechanics Perspective, in: Applied Geotechnics, Spon Press/Taylor & Francis, 2011.
- [54] S.J. Burns, P.T. Piiroinen, K.J. Hanley, Critical time step for DEM simulations of dynamic systems using a hertzian contact model, *Internat. J. Numer. Methods Engrg.* 119 (5) (2019) 432–451.
- [55] K. Arakawa, Dynamic sliding friction and similarity with Stoke’s law, *Tribol. Int.* 94 (2016) 77–81.

- [56] S. Zhu, X. Liu, Q. Cao, X. Huang, Numerical study of tire hydroplaning based on power spectrum of asphalt pavement and kinetic friction coefficient, *Adv. Mater. Sci. Eng.* 2017 (2017).
- [57] G. Pucci, I. Ho, D.M. Harris, Friction on water sliders, *Sci. Rep.* 9 (1) (2019) 4095.
- [58] W.R. Hamilton, II. On quaternions; or on a new system of imaginaries in algebra, *Lond. Edinb. Dublin Philos. Mag. J. Sci.* 25 (163) (1844) 10–13.
- [59] M.P. Allen, D.J. Tildesley, *Computer Simulation of Liquids*, Oxford university press, 1989.
- [60] D. Baraff, An introduction to physically based modeling: Rigid body simulation I—unconstrained rigid body dynamics, in: *SIGGRAPH Course Notes*, 1997.
- [61] M. Kleppmann, *Simulation of Colliding Constrained Rigid Bodies*, University of Cambridge, Cambridge, 2007.
- [62] S.A. Whitmore, Closed-form integrator for the quaternion (euler angle) kinematics equations, Google Patents, US Patent 6,061,611, 2000.
- [63] C.F. Karney, Quaternions in molecular modeling, *J. Mol. Graph.* 25 (5) (2007) 595–604.
- [64] P. Betsch, R. Siebert, Rigid body dynamics in terms of quaternions: Hamiltonian formulation and conserving numerical integration, *Internat. J. Numer. Methods Engrg.* 79 (4) (2009) 444–473.