VIII International Conference on Particle-Based Methods
PARTICLES 2023
U. Perego, M. Cremonesi, A. Franci (Eds)

FAST POINT-TO-MESH DISTANCE COMPUTATION TECHNIQUE BASED ON CELL LINKED LIST FOR POLYGON-WALL BOUNDARY IN MOVING PARTICLE SEMI-IMPLICIT METHOD

MATHEUS C. B. TEIXEIRA[1], LUCAS S. PEREIRA[1], RUBENS A. AMARO JR. [2] AND LIANG-YEE CHENG[1]

[1] Department of Construction Engineering at Polytechnic School
University of São Paulo
São Paulo, SP, Brazil
e-mail: matband@usp.br; lucas_pereira@usp.br; cheng.yee@usp.br

[2] Institute of Mathematics and Computer Sciences (ICMC)
University of São Paulo
São Carlos, SP, Brazil
email: rubens.amaro@usp.br

Abstract. The simulation of fluid-structure interaction (FSI) problems usually involves many degrees of freedom, and a considerable number of particles is generally required to model both fluid and solid domains. In relation to the modeling of solid walls by particles, the use of triangular meshes provides more efficient and smoother representation of complex-shaped solid surfaces as well as the straight coupling between particle and mesh-based methods, which is suitable for FSI applications. However, in the particle-based simulations with solid boundaries modeled by mesh, the computation of the particle-mesh distances is a critical time-consuming task, and a fast technique is of major importance. Taking advantage of the cell linked list structure widely adopted for fixed-radius neighborhood search algorithms in particle methods, we proposed a Fast Point-to-mesh Distance computation technique based on Cell linked list (FPDC). Alongside this new technique, a particle-polygon wall contact model was introduced to enable simulations of the collision between the surface of the moving bodies and fixed wall represented, respectively, by particles and mesh. The results show that the proposed technique provides a significant processing time speedup and can be used for practical large-scale problems.

Keywords: CFD, Fluid Solid Interaction, Hybrid Particle Mesh Method.

1 INTRODUCTION

Lagrangian particle-based methods, such as the Smoothed Particle Hydrodynamics (SPH) [1, 2] and Moving Particle Semi-implicit (MPS) [3], are well-suited for simulating problems involving fluid-structure interaction (FSI). Traditionally, both the fluid and solid domains are modeled as particles and the interaction between fluid and solid surfaces is straightforward. However, in case of complex-shaped bodies, high resolution particle-based wall modeling is generally required for an accurate and smooth surface representation. To overcome this issue, an alternative approach is to use polygonal meshes to represent the surface of the solids, which enables the modeling of complex-shaped solid surfaces while limiting computational costs. Triangular meshes can also be adopted for the coupling between particle and mesh-based methods, such as WCMPS and FEM reported by the authors [4].

In such hybrid particle-mesh approach, the interaction between fluid particles and solid surface meshes depends on the particle to mesh distance. Computation of these distances is a time-consuming and might be a critical task when naively performed. Hence, the development of a fast algorithm for the particle-mesh distance computation is of major importance.

Within this context, the Fast Point-to-mesh Distance computation technique based on Cell linked list (FPDC) is proposed herein to speed up the computation of particle-mesh distances. FPDC takes advantage of the cell linked list structure [5], a widely used structure for neighborhood search algorithms of the particle-based simulations. In this context, it is employed to narrow the domain of the search between the fluid particles and the facets of the mesh.

In this work, the FPDC was implemented and tested in an in-house projection based MPS framework [6]. Nevertheless, it can be easily adapted in other incompressible or weakly compressible particle-based methods, e.g., SPH, incompressible SPH (ISPH) [7] or weakly compressible MPS (WC-MPS) [8]. The efficacy of the proposed FPDC technique is investigated by comparing its performance in point-mesh distance computation and fluid-solid interaction simulation against the Straightforward Method (SM) and the Fastest Closest Points in the West (FCPW) method.

The structure of the present work is as follows: in the next section, the numerical method and the proposed FPDC technique are briefly described. Subsequently, the performance of the FPDC is analyzed. After that, the technique is implemented on a MPS framework and validated using the experimental results of dam breaking hitting a fixed block and stacked cubes. Finally, the conclusions are summarized.

2 NUMERICAL METHODS

In the present study, the fully Lagrangian particle based MPS method was adopted [3] and implemented within an in-house framework [6]. For the neighborhood particle search, a cell linked list [5] structure was utilized. Regarding the boundary conditions, the detection of the free surface is performed using the neighborhood particles centroid deviation (NPCD) technique described in [9]. The traditional particle modeling of the moving solid bodies based on a row of wall particles and two rows of dummy particles combined with the mass proprieties of the solid bodies [5] was adopted. On the other hand, for the surfaces of the fixed walls, instead of the traditional particle-based wall modeling, the Explicitly Represented Polygon (ERP) wall boundary technique [10] was adopted and the surfaces were represented by a triangular mesh. In the ERP technique, the contribution of the polygon meshes to the particle number density and the discrete approximation of the differential operators (gradient, divergence, and Laplacian) depends on the assessment of the distances between the fluid particles to the triangular facets of the mesh. The distance of wall particles of the moving bodies to triangular mesh is also used in the solid contact model adapted from [11]. In this way, the

Fast Point-to-mesh Distance computation technique based on Cell linked list (FPDC) is proposed here to speed up the computation, and it is explained in the following.

2.1 Fast Point-to-Mesh Distance based on Cells (FPDC) technique

The computation of the distance between particles and the nearest triangular facet within its neighborhood is a costly task, particularly when the number of particles and triangles is large, due to its combinatorial nature. Considering the compact support delimited by the effective radius of the particle interaction models, an alternative for efficient computing of the distances is to narrow the search of the triangular facets within the effective radius of a particle using the cell linked list approach [5], which is a technique widely used in the particle-based simulations for neighbor particle search. In this approach, the computational domain is partitioned in a uniformly rectangular grid and only the cell containing the particle and its surrounding cells are considered as the search domain.

Similar to the neighbor particle search, the algorithm of FPDC to speed up the particle-mesh distance calculation is as follows. At first, a loop through all triangular facets of the mesh is made to assign the identifiers (IDs) of the facets into the cell linked list. Like the particles, a cell can contain more than one facet. However, different from the particles that belong to only one cell, a single triangular facet may belong to one or more cells. The assignment of triangular facets to their respective cells is accomplished by detecting intersections between triangles and cells using the Separating Axis Theorem (SAT). For the mesh of a fixed wall, this setup task is performed once at the beginning of the simulation.

After the cell linked list is created, the particle-mesh distance is computed only between the particle and the facets that are in same cell or in neighboring cells from the particle. This distance calculation employs the barycentric coordinates approach for improved efficiency. The particle's position is projected onto the plane of the triangle, resulting in a Voronoi region. If the projection lies within one of the vertex regions, the nearest point on the triangle to the particle is the vertex itself. If it lies within an edge region, the point on the edge closest to the projected point on the plane is selected. Otherwise, the saved projection point is used.

2.2 Contact between the moving bodies and fixed walls

In the present study, since the fixed wall are represented by triangular mesh and the moving bodies are modeled using traditional particle-based modeling, the solid-solid contact model proposed by [11] was adapted to compute the contact between the moving bodies and the fixed walls. Instead of classification of the wall particle as vertex, edge or face depending on its location, all the wall particles of the moving bodies are treated without distinction and only the minimal particle-mesh distance, which is computed using the proposed FPDC technique, is employed to compute the particle-mesh contact distances.

3 PERFORMANCE IN POINT-TO-MESH DISTANCE COMPUTATION

The numerical performance of FPDC is firstly investigated focusing exclusively on the performance for particle-to-mesh distance computation. For this purpose, some simple and well-known body geometries are considered, and the computations of the minimal distance considering different mesh resolutions, i.e., with M particles and N triangles, were carried out. Beside the results of proposed FPDC, the processing times of the Straightforward Method (SM) and the state-of-the-art library Fastest Closest Points in the West (FCPW) method [12] were obtained and used as references for the analysis.

In the SM approach, the distance between a particle (point) and the closest facet is computed by sweeping all facets of the mesh. Only the shortest particle-to-facet distance is considered and if it is smaller than the interaction radius, the facet is considered being in the neighborhood of the particle and the computation of the contributions from the mesh is done using the distance.

In contrast, the FPDC and FPCW algorithms are divided in two parts: setup and computation of particle-mesh distances. In the FPDC, the setup consists of assignment of the triangular facets identifiers into the cell linked list, as detailed in Section 2.1. In the FCPW, the setup consists of building the Bounding Volume Hierarchy (BVH). In case of static meshes, i.e., fixed walls, the setup is executed only once at the beginning of the simulations, while the computation of particle-to-mesh distances is carried out in each time step. Thus, to compare the processing-time from the three approaches, the setup time of FPDC and FCPW are disregarded.

Table 11 shows the processing-times to compute the required particle-to-mesh distances of 10^6 particles in random positions and the meshes of five geometries (cube, tube, Moai, oblong and Bunny with different number of triangular facets) obtained by SM, FCPW and FPDC approaches. For the current analysis, the ratio between particle and mesh resolutions of about $H/l_0 = 30$ was considered, where H is the smallest of the main dimensions of the facets including its width, height, and length, while l_0 is the initial distance between particles. From the table, the proposed FPDC results in a speedup of up to 80x in relation to the SM, which is faster than most of the FCPW results.

Table *1*. Comparison of processing-times (including setup and the required particle-to-mesh distance calculations) obtained using SM, FCPW and proposed FPDC considering five different geometries.

Geometry	**Number of facets**	**SM [s]**	**FCPW [s]**			**FPDC [s]**		
		Particle to mesh distance	**Setup**	**Particle to mesh distance**	**Speedup**	**Setup**	**Particle to mesh distance**	**Speedup**
Cube	12	0.03	0.00	0.12	0.3x	0.03	<u>**0.01**</u>	3.0x
Tube	1202	2.26	0.00	0.07	32.3x	5.50	<u>**0.05**</u>	45.2x
Moai	3012	7.27	0.00	0.25	29.1x	0.78	<u>**0.12**</u>	60.6x
Oblong	4996	9.55	0.00	0.20	47.8x	1.29	<u>**0.12**</u>	79.6x
Bunny	30338	77.00	0.01	<u>**0.33**</u>	233.3x	5.28	1.19	64.7x

Intel® CoreTM i7-11800H, 8 cores (16 threads) at 2.3 GHz

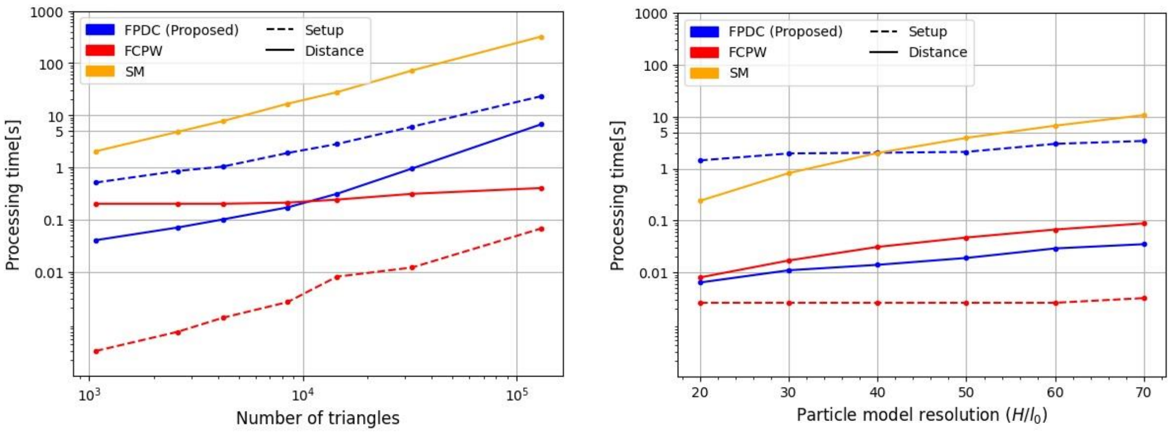
Figure 1 shows the processing time of the SM, FCPW and proposed FPDC for the oblong geometry considering different mesh resolutions and quantity of particles.

In Figure 1a, the number of the particles is fixed to 10^6, and resolution of the mesh of the oblong geometry ranges from 10^3 to 10^5 triangular facets. Compared to FCPW, the efficiency of particle-to-mesh distance computation of FPDC is more sensitive and proportional to the increase of the number of triangular facets. Nevertheless, FPDC performs better for coarser meshes with 10^3 to 10^4 triangles.

On the other hand, Figure 1b, shows the processing times concerning the mesh of the oblong geometry with 8460 triangular facets. The particle model resolution varies from H/l_0 varying from 20 to 70. The results show that the prosed FDPC perform better than FCPW for this range of H/l_0, and the advantage of FDPC is more evident for higher H/l_0 relation.

In summary, the computational cost of FDPC is lower than FCPW for larger number of particles and smaller number of mesh facets. It is relevant to point out that this is the case of the hybrid particle-mesh modeling because while the advantage of the mesh modeling of the solid surface is the adoption of mesh facets much larger than the particles (large H/l_0 ratio), the number of mesh facets required to model the surface increases proportionally to N^2, while the number of the particles increases proportionally to N^3, where N is the resolution of the model.

Finally, according to Figure 1, the setup time of FDPC is much higher than FCPW. This is not a problem in case of fixed meshes because the setup is performed only once at the beginning of the simulation. However, it is a big issue in case of moving of deformable surface meshes. Since improvement of the setup algorithm can still be made as well as the application of parallel processing for the setup task, it is an interesting topic for a future study.



(a) Effects of the mesh resolution (b) Effects of the particle resolution

Figure 1. Comparison between the processing-times of the SM, FCPW and FPDC.

4 PERFORMANCE IN THE HYDRODYNAMIC SIMULATIONS

The proposed FPDC technique for particle-mesh distance calculation was implemented in an in-house MPS framework and, in this section, the performances of the technique were evaluated considering two cases: a 3D dam breaking hitting a fixed block and a 3D dam breaking hitting six heavy cubic solids in a pyramid formation, which also interact with each other and with the fixed walls.

4.1 3D dam breaking hitting fixed block

The experimental 3D dam breaking hitting a fixed block carried out by Kleefsman et al. [13] was chosen as for the benchmark test. The geometry and main dimensions, as well as the initial conditions are shown in Figure 2a. The positions of the pressure sensors position, which are located on the fixed block, are presented in Figure 2b. The fluid density and viscosity are $\rho = 1000$ kg/m³ and $\nu = 10^{-6}$ m²/s, respectively. The acceleration of gravity is $g = 9.81$ m/s².

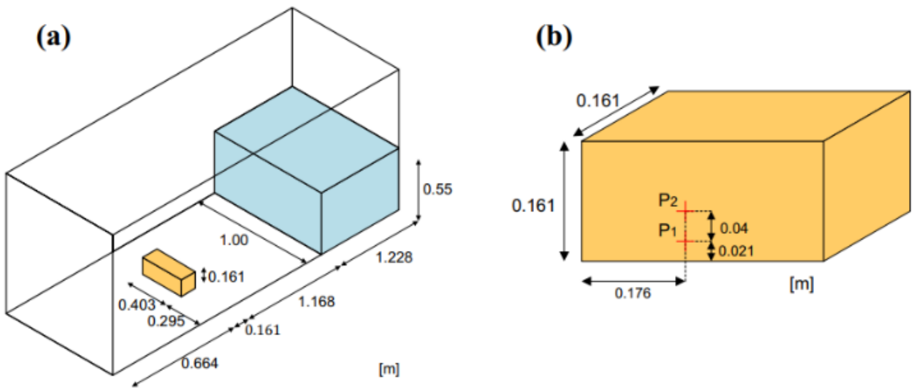


Figure 2. Main dimensions of the 3D dam breaking hitting a fixed block performed by Kleefsman et al. [13]. (a) Initial geometry and conditions, and (b) pressure sensor position on the block (in meters). Reprinted from Amaro et al. [11].

In the simulations, the fluid is modeled by particles, while the tank walls as well as the fixed block are modeled by triangular meshes. Four different particle resolutions were considered $L/l_0 = 4$, 8, 16 and 21, where L is the fixed block height ($L = 0.161$m). Table 2 presents the number of particles and numerical parameters adopted in the simulations of each resolution. The number of triangular facets in the mesh of the tank and fixed block is 28 for all resolutions. The simulation duration was 6s.

Table 2. Numerical parameters and number of particles of the 3D dam breaking hitting fixed block.

Resolution L/l_0	**4**	**8**	**16**	**21**
Initial distance between particles l_0 [mm]	40	20	10	7.5
Number of particles	10850	82350	676500	1592276
Fluid time step Δt [ms]	2.5	1.0	0.5	0.25

Figure 3 illustrates snapshots of the simulation considering the fine resolution $L/l_0 = 21$ at three instants. At $t = 0.5$s, the wave front impacts and flows around the fixed block, see ***Figure 3***a. Subsequently, the wave front overtops the fixed block and reaching the downstream wall, when a runup is present, see ***Figure 3***b. Finally, the flow returns above the fixed block, see ***Figure 3***c.

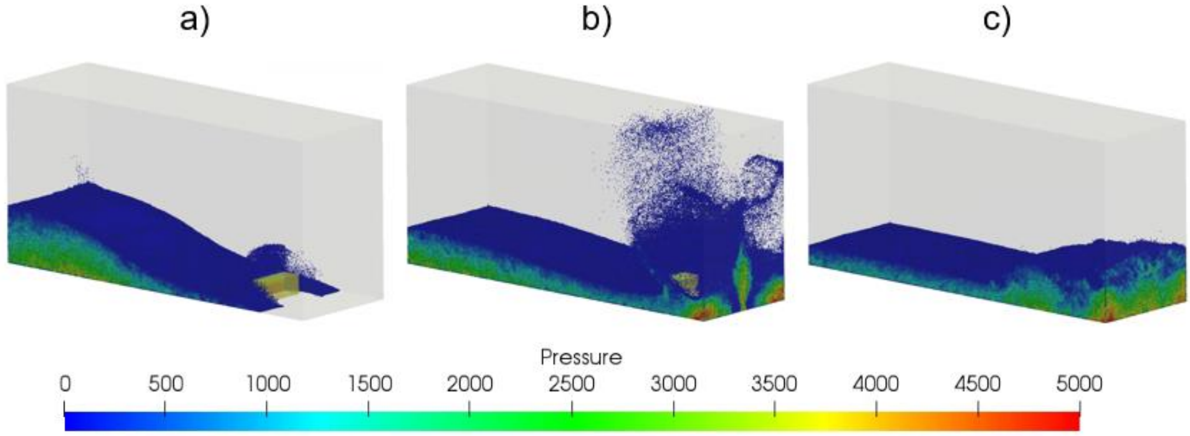


Figure 3. Snapshots of the simulations of the 3D dam breaking hitting fixed block for three instants: (a) $t = 0.5s$, (b) $t = 3.25s$ and (c) $t = 4.75s$. The color scale of fluid particles represents their pressure magnitude.

The computed pressure time series at the sensor P_1 and P_2 are shown in Figure 4 together with the experimental results. The pressure computed by MPS with the fixed wall modeled by triangular mesh was calculated using the weighted average value of all fluid particles within the effective radius of the sensor and based on the weight function ($r_e = 2.1l_0$). The overall trend of computed pressure time series agrees well for resolution higher than $L/l_0 = 8$. As the computed pressure using MPS is an averaged value, the sharp impact peaks are slightly lower than the experimental ones and they are better captured as the resolution increases. The slight delay between instants $t = 4.5$ s and $t = 5.0$ s is also reduced as the resolution increases.

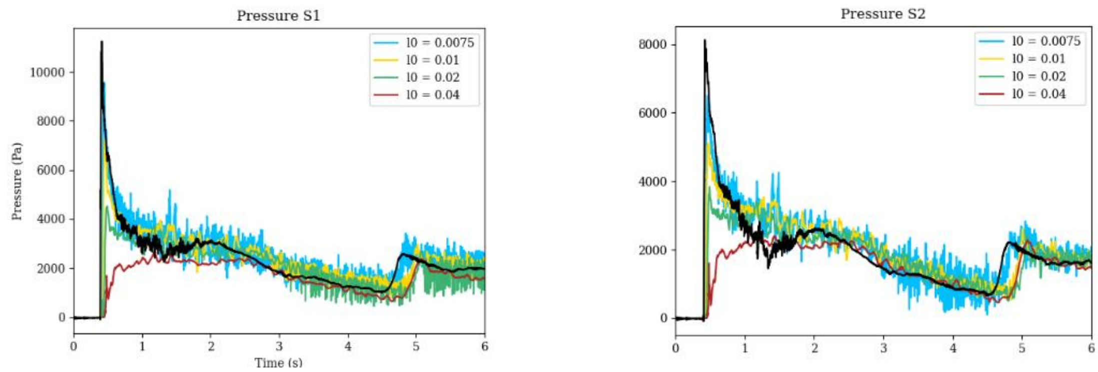


Figure 4. Time series of pressure computed by the present MPS and experimentally measured by Kleefsman et al. [13] at sensors P_1 and P_2.

Table 3 presents the processing time of the entire 6 s simulation for the four resolutions and using either SM or FPDC techniques to compute the particle-mesh distance. The proposed FPDC technique results in a speedup between 2x and 3x in comparison with the SM.

Table 3. Processing time using the SM and FPDC approaches for 3D dam breaking hitting fixed block.

Resolution	**Processing time* [h]**		**Speedup**
L/l_0	**SM**	**FPDC**	
4	0.034	0.016	**2.13x**
8	0.6	0.23	**2.61x**
16	19.1	6.67	**2.86x**
20	96.45	43.2	**2.23x**

*Intel® Xeon® Processor E5-2680 v2 2.80GHz, 10 Cores (20 Threads)

4.2 3D dam breaking hitting stacked cubes

The experimental results of 3D dam breaking event hitting stacked cubic solids provided by Canelas et al. [14] were adopted to analyze the performance of the particle-polygon wall contact model described in Section 2.2. The main dimensions and initial arrangement of the six heavy cubes in a triangle or pyramid formation are displayed in Figure 5. The central cubes were aligned with the center line of the tank, and cubes of the same level were separated by gaps of 0.05 m and the side length of the cubes is $S = 0.15$ m.

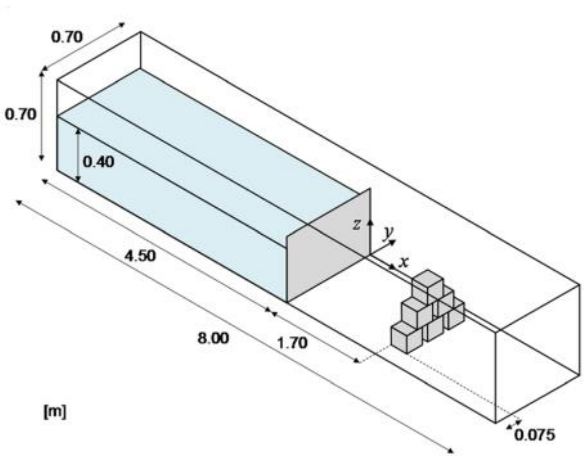


Figure 5. Main dimensions of the 3D dam breaking experiment with six cubes in triangle (pyramid) formation performed by Canelas et al. [14]. Reprinted from Amaro et al. [11].

In the simulations, the fluid, the floodgate and the six cubic solids were modeled by particles, while the tank walls were modeled by triangular meshes. When the simulation began, the floodgate was lifted with a constant velocity of 1.9 m/s and it was completely opened after 0.21 s. The fluid density and kinematic viscosity are $\rho = 1000$ kg/m³ and $\nu = 10^{-6}$ m²/s, respectively. The gravity acceleration is $g = 9.81$ m/s². Table 4 shows the material properties and numerical parameters used in the simulations. The material properties, such as density, mass, Poisson's ratio, and Young's modulus were provided by [14], while the collision coefficient as well as the static friction coefficient were obtained though numerical calibration by [11].

Table 4. Material properties and numerical parameters of the 3D dam breaking hitting stacked cubes.

Material	Density ρ [kg/m³]	Mass m [kg]	ν	E [GPa]	Collision coef. ξ_n	Static friction coef. μ
Cube	800	2.70	0.3	3	0.05	0.2
Wall	∞	∞	0.3	210	–	0.05

Three resolutions were adopted for the 3D dam breaking hitting stacked cubes, namely S/l_0 = 12, 15 and 20, where S is the length of the side of the cubes (S = 0.15 m). Table 5 depicts the number of particles and time step for each simulation resolution. The duration of the simulations was 2 seconds. The number of triangular facets in the mesh of the tank is 12 for all resolutions.

Table 5. Numerical parameters of the 3D dam breaking hitting stacked cubes.

Resolution S/l_0	**12**	**15**	**20**
Initial distance between particles l_0 [mm]	12.5	10	7.5
Number of particles	656432	1286088	3023871
Fluid time step Δt [ms]	0.5	0.5	0.25

Figure 6 exhibits top-view snapshots of the experiments [14] and simulations from the current study using initial distances between particles l_0 = 10 and 7.5 mm. At t = 0.95s, the wave front hits the cubes, resulting in displacement from the original position, see ***Figure 6***a. Subsequently, the cubes at the stack's base are carried by the wave, while the remaining cubes fall at t = 1.15 s (see ***Figure 6***b). Finally, at t = 1.45 s, the cubes are separated from each other, with similar positions in both experimental and numerical results (see ***Figure 6***c). In general, the MPS simulations are in good agreement with the experimental results, particularly for the case with initial distance between particles of l_0 = 7.5 mm (S/l_0 = 20).

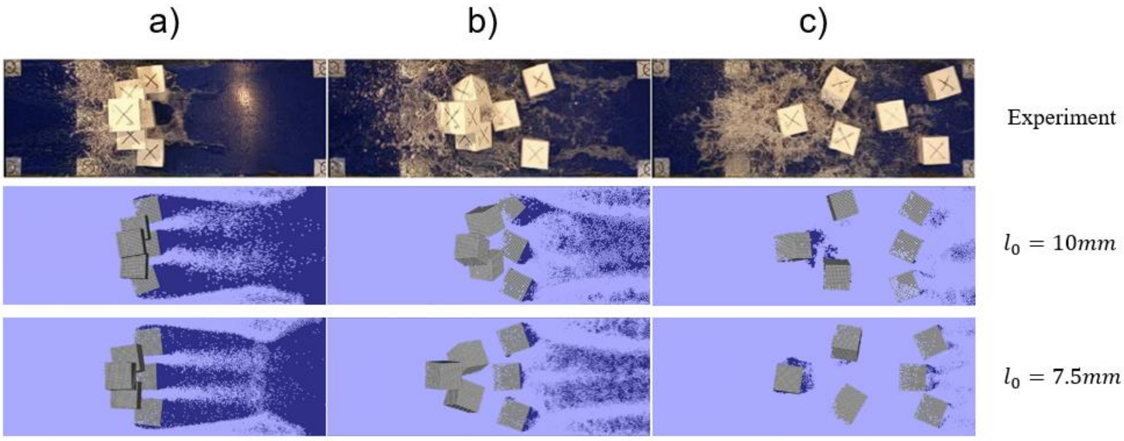


Figure 6. Snapshots of the 3D dam breaking hitting stacked cubes of the experiments performed by Canelas et al. [14] and simulations of the present study for three instants: (a) $t = 0.95\ s$, (b) $t = 1.15\ s$ and (c) $t = 1.45\ s$.

Figure 7 illustrates the CG positions of the top cube along longitudinal and vertical directions during the dam breaking event. When the wave front reaches the stacked cubes at t = 0.8s, the top cube moves slowly along the longitudinal direction (see ***Figure 7***a). After t = 1.0s, the top cube falls from the original vertical position, reaching the tank bottom after approximately 1.3s (see ***Figure 7***b). Subsequently, the longitudinal displacement of the cube increases quickly (see ***Figure 7***a). The cube remains in contact with tank bottom until the end of the simulations. In general, the computed positions show a similar trend in comparison with the experimental one. As the resolution increases, the time series of the longitudinal displacement is in better agreement with the experimental results.

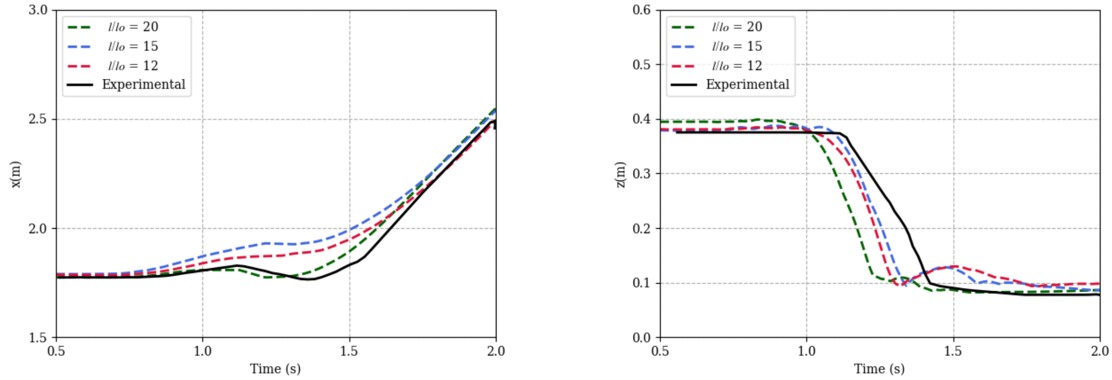


Figure 7. Time series of the top cube position along the (a) longitudinal and (b) vertical directions.

The processing times of the 2 s dam breaking event simulation using the SM and the proposed FPDC technique to compute the particle-mesh distance are given in Table 6. In general, a speedup of about 2 times in relation to the SM were achieved by the proposed FPDC technique.

Table 6. Processing times using the SM and FPDC approaches for 3D dam breaking hitting stacked cubes simulations.

Resolution L/l_0	**Processing time* [h]**		**Speedup**
	SM	**FPDC**	
15	10.37	4.67	**2.2x**
20	56.03	26.22	**2.1x**

*Intel® Xeon® Processor E5-2680 v2 2.80GHz, 10 Cores (20 Threads)

5 FINAL CONSIDERATIONS

In the present work, a technique to improve the calculation of particle-mesh distance is proposed for the particle-based fluid dynamics simulations with fixed wall modeled by triangular mesh. The Fast Point-to-mesh Distance computation technique based on Cell linked list (FPDC) takes advantage of the cell linked list structure to narrow the search domain and speed up the processing time. Moreover, a particle-polygon wall contact model was introduced to enable simulations of the collision between the surface of the moving bodies and fixed wall, which are represented, respectively, by particles and mesh.

At first, the numerical performance of the FPDC technique was investigated by comparing its processing time against the results of Straightforward Method (SM) and Fastest Closest Points in the West (FCPW). As a result, when computing the particle-mesh distances, a speedup of up to 80x on SM was achieved by FPDC. In comparison with FCPW, the FPDC performs better for larger number of particles and smaller number of mesh facets, which is a situation very suitable to the hybrid particle-mesh modeling.

Subsequently, the proposed FDPC technique was implemented in an in-house MPS simulation framework and validated using two benchmark cases involving 3D dam breaking on a fixed block and on a stacked cube. The results shows that the particle-polygon wall contact model is effective and about 2 times speedup were achieved in relation to the simulations based on SM.

The proposed FDPC technique is well-suited to cases with fixed meshes, in which the setup is performed only once at the beginning of the simulations. Aiming the simulation of moving or deformable bodies modeled by polygon meshes the improvement of the setup algorithm of FDPC is a topic for future study.

ACKNOWLEDGMENTS

The authors would like to thank the Coordination for the Improvement of Higher Education Personnel – CAPES and University of São Paulo, respectively, for the doctoral and undergraduate research scholarships (PUB). R.A. Amaro Jr gratefully acknowledges the financial support received from the São Paulo Research Foundation FAPESP/CEPID/CeMEAI grant 2021/11429-4.

REFERENCES

[1] R. A. Gingold and J. J. Monaghan, “Smoothed particle hydrodynamics: theory and application to non-spherical stars,” *Mon Not R Astron Soc*, vol. 181, no. 3, pp. 375–389, Dec. 1977, doi: 10.1093/mnras/181.3.375.

[2] L. B. Lucy, “A numerical approach to the testing of the fission hypothesis,” *Astron J*, vol. 82, no. 12, pp. 1013–1024, 1977.

[3] S. Koshizuka and Y. Oka, “Moving particle semi implicit method for fragmentation of incompressible fluid,” *Nuclear Science and Engineering*, vol. 123, no. 3, pp. 421–434, 1996.

[4] R. A. Amaro Junior, A. Gay Neto, and L. Cheng, “Three-dimensional weakly compressible moving particle simulation coupled with geometrically nonlinear shell for hydro-elastic free-surface flows,” *Int J Numer Methods Fluids*, vol. 94, no. 8, pp. 1048–1081, Aug. 2022, doi: 10.1002/fld.5083.

[5] Z. Yao, J.-S. Wang, and M. Cheng, “Improved O(N) Neigh bor List Method Using Domain Decomposition and Data Sorting,” *2004 NSTI Nanotechnology Conference and Trade Show - NSTI Nanotech 2004*, vol. 2, Dec. 2003.

[6] M. M. Tsukamoto, L.-Y. Y. Cheng, and K. Nishimoto, “Analytical and numerical study of the effects of an elastically-linked body on sloshing,” *Comput Fluids*, vol. 49, no. 1, pp. 1–21, 2011, doi: 10.1016/j.compfluid.2011.04.008.

[7] S. Shao and E. Y. M. Lo, “Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface,” *Adv Water Resour*, vol. 26, no. 7, pp. 787–800, Jul. 2003, doi: 10.1016/S0309-1708(03)00030-7.

[8] A. Shakibaeinia and Y. C. Jin, “A mesh-free particle model for simulation of mobile-bed dam break,” *Adv Water Resour*, vol. 34, no. 6, pp. 794–807, 2011, doi: 10.1016/j.advwatres.2011.04.011.

[9] M. M. Tsukamoto, L. Y. Cheng, and F. K. Motezuki, “Fluid interface detection technique based on neighborhood particles centroid deviation (NPCD) for particle methods,” *Int J Numer Methods Fluids*, vol. 82, no. 3, pp. 148–168, 2016, doi: 10.1002/fld.4213.

[10] N. Mitsume, S. Yoshimura, K. Murotani, and T. Yamada, “Explicitly represented polygon wall boundary model for the explicit MPS method,” *Comput Part Mech*, vol. 2, no. 1, pp. 73–89, 2015, doi: 10.1007/s40571-015-0037-8.

[11] R. A. Amaro Junior, L.-Y. Cheng, and P. H. S. Osello, “An improvement of rigid bodies contact for particle-based non-smooth walls modeling,” *Comput Part Mech*, vol. 6, no. 4, pp. 561–580, Oct. 2019, doi: 10.1007/s40571-019-00233-4.

[12] Rohan Sawhney, Max Slater, Ruihao Ye, Keenan Crane, and Johann Korndoerfer, “FCPW: Fastest Closest Points in the West.,” *Online*. https://github.com/rohan-sawhney/fcpw (accessed Sep. 19, 2023).

[13] K. M. T. Kleefsman, G. Fekken, A. E. P. Veldman, B. Iwanowski, and B. Buchner, “A Volume-of-Fluid based simulation method for wave impact problems,” *J Comput Phys*, vol. 206, no. 1, pp. 363–393, 2005, doi: 10.1016/j.jcp.2004.12.007.

[14] R. B. Canelas, A. J. C. Crespo, J. M. Domínguez, R. M. L. Ferreira, and M. Gómez-Gesteira, “SPH-DCDEM model for arbitrary geometries in free surface solid-fluid flows,” *Comput Phys Commun*, vol. 202, pp. 131–140, May 2016, doi: 10.1016/j.cpc.2016.01.006.