Proceedings of the Twenty-first (2011) International Offshore and Polar Engineering Conference
Maui, Hawaii, USA, June 19-24, 2011

ISBN 978-1-880653-96-8 (Set); ISSN 1098-6189 (Set); www.isope.org

Numerical Simulation of Sphere Water Entry Problem Based on VOF and Dynamic Mesh Methods

Zhirong Shen and Decheng Wan
State Key Laboratory of Ocean Engineering, School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University
Shanghai, China

ABSTRACT

This paper presents a numerical experiment of single low-density sphere falling into the water from the air. A sphere with initial speed in the air was released above the water surface, splashes and waves will be generated when it touched the water; after that, it jumped above water because of buoyancy. The procedure is simulated by the Open source Field Operation and Manipulation (OpenFOAM) based on VOF method and dynamic mesh method. The result of time history of hydrodynamic forces and sphere's motion is given. Validations of grid and time convergence are performed in order to prove the correctness of this method. Results of the velocity field and the pressure distribution around the sphere are discussed.

KEY WORDS: water entry; OpenFOAM; VOF; dynamic mesh; 6DOF

INTRODUCTION

The significance of research on water entry problem is the large damage impact force on ship and ocean structures. In rough sea, large motion of the ocean platforms or ship hulls can usually cause great damage to the structures when they touch and entry the water. This kind of damage sometimes can cause the sink of ocean structure, which will bring huge economic losses.

The problem of sphere water entry is related to large amplitude motion of body in fluid with large free surface deformation. Traditional approaches to solve the problems are usually based on strip theory and other linear methods. Although fast and easily accessible, these methods are not accurate enough and fail to deal with the large motion of body or waves. In this paper, water entry problem is solved by incompressible Navier-Stokes equations based on VOF surface capturing method and automatic deformation dynamic method.

In recent decades, many methods were developed to solve these kinds of problems. Battistin(2003) used a boundary-element formulation with nonlinearities in the free-surface boundary conditions to investigate the water entry of two-dimensional bodies of arbitrary shape. Iafrati(2008) solved the flat plate impact problem by using method of matched asymptotic expansions. Kleefsman(2005) simulated the water impact problem of falling wedge and cylinder by solving an incompressible viscous NS equations with VOF method. Maruzewski(2010) developed a SPH method to simulate the water impact process of a snooker ball. In this paper, an incompressible Navier-Stokes equations based on VOF surface capturing method and automatic deformation dynamic mesh method was applied to simulate the water entry problem of a sphere. The results include the transient behavior of hydrodynamic forces and sphere's motion, and the viscous flow and surface pressure distribution of the sphere. The key points are the initial stage of the water entry movement including impact forces that may cause the greatest damage to the object.

Although there are many research works on the water entry problems, the experimental data or computational results for low-density floating sphere are very rare. Therefore, the validation mainly depends on the convergence study of this method itself.

For the surface capturing method, the most popular methods are level-set method and VOF method. Level-set was first introduced by Osher(1988), where a distance function ϕ was introduced to denote the distance to the surface. This method can treat highly distorted interfaces, but has one major shortage: the lack of mass conservation. In this paper, we choose another surface capturing method, Volume of Fluid (VOF). VOF method, which was first introduced by Hirt(1981), uses a function named volume of fraction between zero and one to indicate the fractional volume of a cell that filled with a certain fluid. Compared with level-set method, VOF method is mass conservation. Now it has been widely used by many CFD softwares and codes. And it was realized in OpenFOAM by Rusche(2003).

For the moving boundary problem, in the recent decades, many methods have been developed to solve this problem. One popular method is overset grid. In this method, one or more body fitted grids, which wrap up objects in it, are embedded into a Cartesian background grid. Therefore, the body fitted grids can translate and rotate in any direction without changing the background mesh. Overset method was successfully applied to ship seakeeping and maneuverability problems by Carrica(2007; 2008). Another method is fictitious boundary method (FBM) based on finite element method introduced by Wan(2006; 2007).

The advantage of FBM is that the objects in the fluid, which can be assigned different shape and size, can move freely through the computational mesh for the fluid part, which does not change with time. The last and the simplest method is the dynamic mesh, where mesh nodes move and the shapes of mesh deform according to the motion of object. Dynamic mesh has been applied to many CFD softwares and codes because it is easy to handle. Therefore, in present work, the moving boundary problem is based on the dynamic mesh.

In recent years, the Open source Field Operation and Manipulation (OpenFOAM) C++ libraries provide users open source codes for developing new CFD methods. Since the codes are open and free to everyone, users can directly use the code or develop new solver based on the original one to solve different problems. Besides, OpenFOAM is also a good environment for user to develop new methods because it provides numerous CFD related libraries and utilities, such as basic tensor mathematics, essential algebra solvers, different widely used turbulence models, dynamic mesh, large-scale parallel computing and many pre- and post-processing utilities. With the help of OpenFOAM, users can directly solve engineering problems or develop new method based on their ideas, instead of devoting time and energy in developing the basic elements of CFD codes. In this paper, the solver is based on a code named shipFoam (an extended solver of OpenFOAM provided by OpenFOAM Ship Hydrodynamics Special Interest Group (SIG)) with rewriting some modules such as force and moment calculation, 6DOF motion solver and so on.

NUMERICAL METHOD

Governing Equations

In this paper, an incompressible, viscous flow is concerned, which can be described by the Navier-Stokes equations and the continuity equation as follows.

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot \left(\rho \mathbf{U}\mathbf{U}\right) = -\nabla p + \rho \cdot \mathbf{g} + \nabla \cdot \left[\mu\left(\nabla \mathbf{U} + \left(\nabla \mathbf{U}\right)^T\right)\right] + f_\sigma + f_s \quad (1)$$

$$\nabla \cdot \mathbf{U} = 0 \quad (2)$$

Where, $\mathbf{U}$ is velocity; p is pressure; μ is dynamic viscosity; $\mathbf{g}$ is gravity acceleration; f_σ and f_s are both source terms, the former is due to surface tension and the latter is added for sponge layer.

Surface Interface

VOF method is applied in present work to deal with the surface interface. The surface tension in the Navier-Stokes equation is defined as:

$$f_\sigma = \sigma \kappa \nabla \alpha \quad (3)$$

Where, σ is the surface tension (in present work is set to $0.07N/m$); κ is the curvature of the interface, given by:

$$\kappa = -\nabla \cdot \left(\frac{\nabla \alpha}{|\nabla \alpha|}\right) \quad (4)$$

α is the volume of fraction, which is equal to one in water and zero in air, indicating cells filled with certain water. The volume of fraction obeys a transport equation:

$$\frac{\partial \alpha}{\partial t} + \left(\mathbf{U} \cdot \nabla\right)\alpha = 0 \quad (5)$$

Pressure-Velocity Solution Procedure

The solution of Navier-Stokes and continuity equation is obtained by applying the pressure-implicit split-operator (PISO) algorithm, which was first developed by Issa(1986), and applied in OpenFOAM by Jasak(1996). The method uses a predictor-corrector method to solve the Navier-Stokes equations by applying the continuity equation. It uses a collocated grid introduced by Rhie(1983) instead of staggered grid to save the memory for additional copies of mesh.

Dynamic Mesh

In this paper, an automatic mesh deformation method in OpenFOAM (Jasak, 2006; 2009), is chosen to solve the mesh motion. This method introduces a diffusivity model to control mesh movement and mesh quality. In present work, the diffusivity is based on the inverse of cell volume, which means that the cells with smaller volume (usually the important cells) will be maintained better quality during the mesh moving.

6DOF Motion Solver

Although only one degree of freedom is taken into account in this paper, a fully 6DOF motion solver is developed in order to solve much more complicated motion in future work. The details can be found in (Carrica, 2007). In this method, two coordinate systems are used to solve 6DOF equation. One system is earth fixed system and the other is body fixed system. In earth fixed system, $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) = (x, y, z, \phi, \theta, \psi)$ are the translations and rotations of the object, which represent motions of surge, sway, heave, roll, pitch and yaw respectively. Similarly, $\dot{\mathbf{x}} = (\dot{\mathbf{x}}_1, \dot{\mathbf{x}}_2) = (\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$ are the linear and angular velocities of the six types of motion. On the other hand, $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2) = (u, v, w, p, q, r)$ and $\dot{\mathbf{v}} = (\dot{\mathbf{v}}_1, \dot{\mathbf{v}}_2) = (\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r})$ represent velocities and accelerations in body fixed system. The angular velocities in body fixed system, which are $\dot{\mathbf{v}}_2 = (\dot{p}, \dot{q}, \dot{r})$, can be transformed to that of earth fixed system by following equation:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi / \cos\theta & \cos\phi / \cos\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (6)$$

Moreover, the linear velocities in body fixed system can be transformed in a similar way:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} (7)$$

After we establish the connection between two systems, the 6DOF equations can be solved in the following steps:

1. Calculate the hydrodynamic forces and moments in the earth fixed system after the flow is converged.
2. Transforms the forces and moments, including gravity forces and moments, into body fixed system by Eq.(7). Moreover, the following equations need to be solved to calculate the linear and angular acceleration in body fixed system.

$$
\begin{cases}
\dot{u} = X / m + vr - wq \\
\dot{v} = Y / m + wp - ur \\
\dot{w} = Z / m + uq - vp \\
\dot{p} = K / I_x + (I_y - I_z)qr \\
\dot{q} = M / I_y + (I_z - I_x)rp \\
\dot{r} = N / I_z + (I_x - I_y)pq
\end{cases} \tag{8}
$$

Where, m is the mass of the object. I_x, I_y, I_z are principal axes of inertia. X, Y, Z, K, M, N are respectively surge, sway, heave forces and roll, pitch, and yaw moments.

3. Integrate the acceleration to velocities and transform them to earth fixed system by Eq.(6) and Eq.(7).
4. Calculate the velocities of mesh points on object boundary according to the velocities solved in Step.3.And send the boundary points' velocities to the dynamic mesh solver in OpenFOAM.
5. Run the dynamic mesh solver to compute the movements of global mesh points, and update the mesh.
6. This iteration is complete and the next iteration starts.

Sponge Layer

Sponge Layer is implemented to absorb the reflect waves and reduce the influence on sphere's motion. The application of sponge layer is simply add a source term is Navier-Stokes equations (Eq.1):

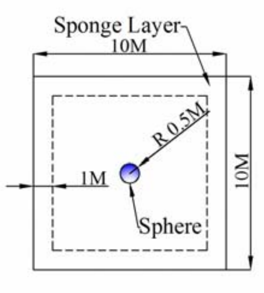
$$f_s = -\rho \theta \mathbf{U} \tag{9}$$

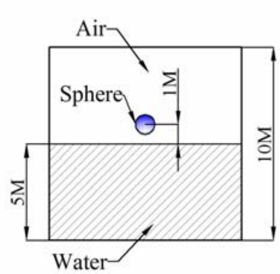
θ can be set to linearly from 0 to 10 where the sponge layer is applied. In other region, θ is equal to 0.

GEOMETRY AND CONDIDTION

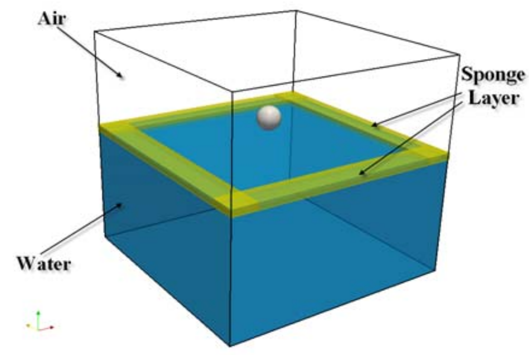
Geometry

A tank with a half of water is the computational domain, as shown in Fig. 1. The length of tank's each edge is equal to 10 meters. A sphere with density of $500 kg/m^3$ is fixed above the surface. The radius of the sphere is 0.5m. The center of gravity of the sphere is placed at 1m above the surface. Sponge Layers are set up on the near wall region to absorb waves produced by the sphere. The width of sponge layer is 1m. At t=0, the sphere will be released with different initial velocities (0m/s, 0.5m/s, 1.0m/s) to observe the different impact forces due to different impact velocity.





(a) Top View (b) Front View



(c) 3D View

Fig. 1 Computational model of this study

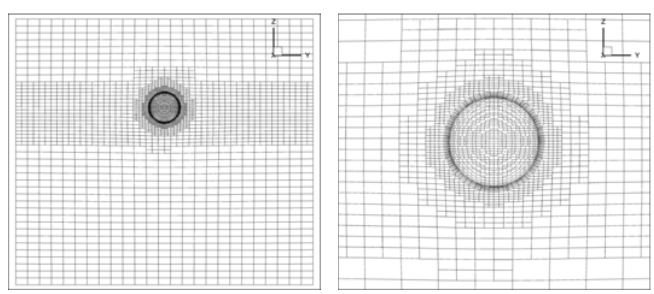
Grid and Time Step Convergence Study

In the grid convergence study, five meshes with different cell numbers are chosen to take grid convergence study, as shown in Table 1. The sizes of meshes range from 86,787 to 1,315,946 cells.

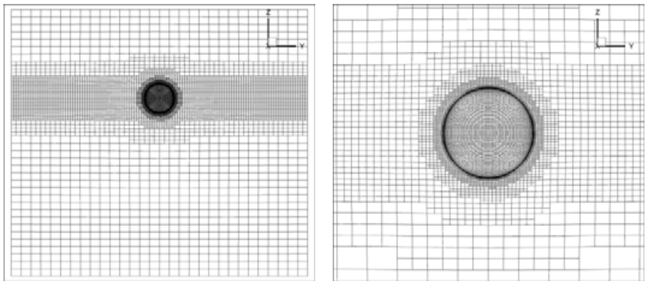
Fig. 2 shows the transverse section of three different meshes. In this paper, all meshes are automatically generated by the snappyHexMesh utility, which is provided by OpenFOAM. The utility generates 3D meshes by split hexahedra cells around object into small cells. The split cells continue to split until reach the level defined by user. The region from z=-0.3 to z=1.8m needs special refinement, because in this region, free surface properly exists. Since the mesh moves according to the motion of sphere and the exact position of free surface interface is unknown, the special refinement region need to be enlarged to ensure that every possible free surface interface is inside this region.

Table 1. Five different meshes for grid convergence study

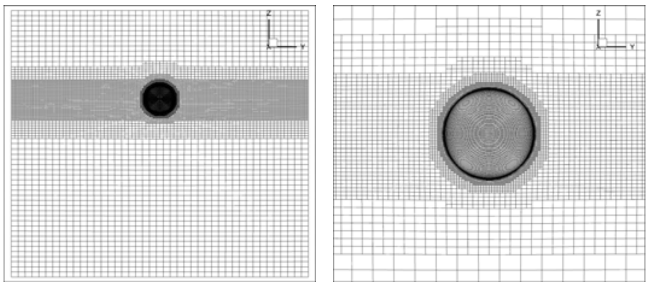
Mesh Name	Mesh I	Mesh II	Mesh III	Mesh IV	Mesh V
Cell Number	86,787	315,545	505,736	786,765	1,315,936



(a) Mesh I (coarsest mesh)



(b) Mesh III (medium mesh)



(c) Mesh V (finest mesh)

Fig. 2. Transverse section of the coarsest, medium and finest meshes

In OpenFOAM, time step is controlled by Courant Number Co, given by:

$$Co = \frac{\delta t|\mathbf{U}|}{\delta x} \tag{10}$$

Where, δt is time step, $\mathbf{U}$ is field velocity, δx is the minimum grid space. We define $MaxCo$ as maximum Co. By specifying the target $MaxCo$, system automatically calculates δt during each time step iteration. In the time step study, we set four different time step sizes based on four different targets: $MaxCo = 0.05$, 0.10, 0.20 and 0.30.

RESULT AND DISCUSSION

Global Overview

At the initial stage, the sphere is fixed in the air above the surface, and released with an initial downward speed. We take the case of initial velocity equal to zero (v=0) as example. As shown in Fig. 3, the downward movement is defined as negative. Fig. 4 shows six snapshots of the sphere's movement. The sphere dropped from the original position (z=0), and at t=0.99s, sphere reached the maximum displacement in the water. Then, it floated up and jumped to the air because of buoyancy force. At t=1.90s, it reached top position in the air, and continued to the water for the gravity force. The down-and-up movement continued but the motion amplitude was diminished under the effect of the damping forces, such as friction. At the end of the computation, the amplitude of displacement was almost reduced to zero. It shows that, the sphere would attain a steady condition and keep statically floating on the surface.

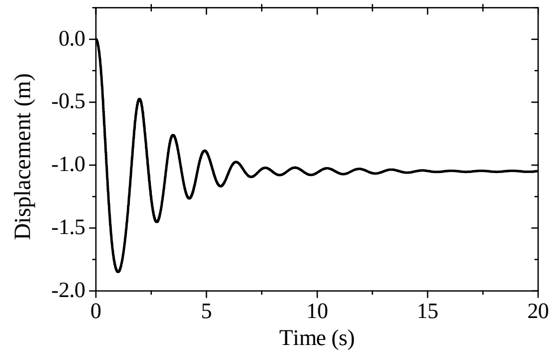
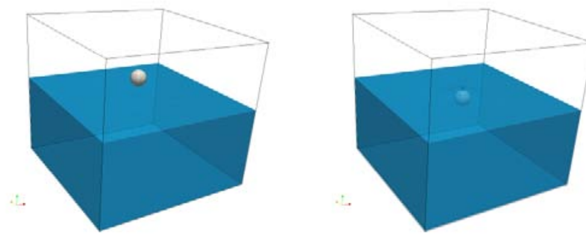
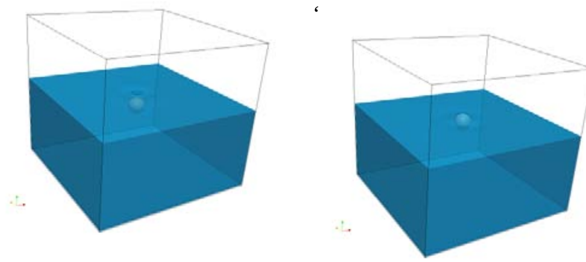


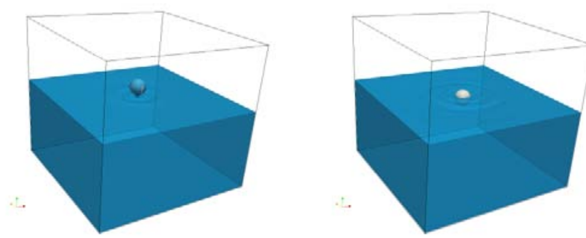
Fig. 3 Transient behavior of Sphere Movement



(a) Initial Stage (t=0) (b) Entering water (t=0.6s)



(c) Max Displacement (t=1.0s) (d) Returning to Air (t=1.5s)



(e) Jump to Air (t=2.0s) (f) Float Condition (t=5.0s)

Fig. 4. Snapshots of Sphere's Movement

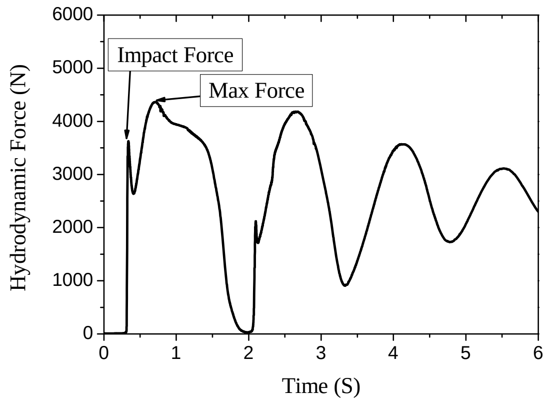


Fig. 5. Time History of Hydrodynamic Forces

Fig. 5 shows the time history of vertical hydrodynamic forces of the sphere. The forces curve, which is similar with that of sphere displacement, shows an up-and-down trend. Right before the sphere reached maximum displacement, the vertical hydrodynamic force reached maximum force (We define the force as max force f_M). Similarly, when the sphere returned to air at about t=2.0s, the forces reduced to nearly zero. Similar with the curve of sphere movement, the amplitude of vertical forces also diminished, which showed that the sphere reached an equilibrium condition. The key point is that before t=0.3s, the force was close to zero because in this moment, the sphere was in the air and the aerodynamic force was quite small compared with hydrodynamic force. However, after that, the vertical forces increased dramatically and reached a peak value in nearly 0.03 seconds, and then dropped to a normal value. The sharp increasing forces were due to the sphere's impact effects. We define it as impact force f_I . The similar phenomenon occurred again at about t=2.03s because the sphere had a second impact, but the force is relatively smaller than the first one. The impact force is quite important for ocean and ship engineering. The sharply increasing force can create large damage to structure. However to capture accurately the impact force is quite difficult. So next, we take the first impact force f_I as validation object. Since the max vertical force f_M is also crucial, so f_M is another object, as shown in Fig. 5.

Mesh Convergence Study

In the mesh convergence study, first, we chose five meshes with different cell numbers as shown in Table 1. The validation objects were the impact force f_I and the max force f_M as shown in Fig. 5. Second, we set three different downward initial velocities: $V = 0.0m/s$, $V = 0.5m/s$ and $V = 1.0m/s$. For each initial velocity, we can get different impact force f_I and max force f_M . Fig. 6 is the result of grid convergence of impact force f_I . The x-axis represents the size of each mesh, and each curve represents different initial velocities. It can be seen that for the coarser meshes, such as Mesh I and Mesh II, impact force f_I was severe underpredicted compared to the finer Mesh. With the increase of cell number, the changes of f_I became relatively smaller. For finer meshes (Mesh VI and Mesh V), the results of f_I were almost consistent. For the medium mesh (Mesh III), the impact force was slightly overestimated in the cases of V=0.5m/s and V=1.0m/s, but consistent in case of V=0.0m/s. For different initial velocity, the larger the initial velocity, the bigger the impact force was.

Fig. 7 is the result of grid convergence of max force f_M . Similar with the result of impact force f_I , coarse meshes usually could not capture the force accurately. For coarser meshes, such as Mesh I and II, the max force f_M was severely overpredicted. However, for the medium and finer meshes, the results were almost the same. With the increase of mesh size, the differences in max force f_M were becoming smaller, especially for the last three meshes (Mesh III, IV and V). Just like the results of impact force f_I , these results also showed good grid convergence of this method. For the part of initial velocity, to our surprise, max force f_M with different initial velocity were very close to each other. It showed that the result of max force f_M might be irrelevant to the initial velocity. It was reasonable due to the fact that sphere with large velocity might lose more velocity when entering the water because larger velocity would induce larger resistance force.

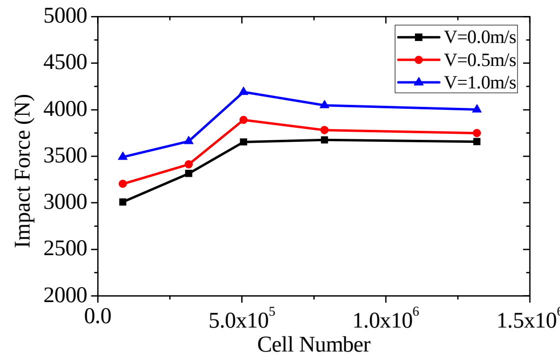


Fig. 6. Grid Convergence of Impact Force

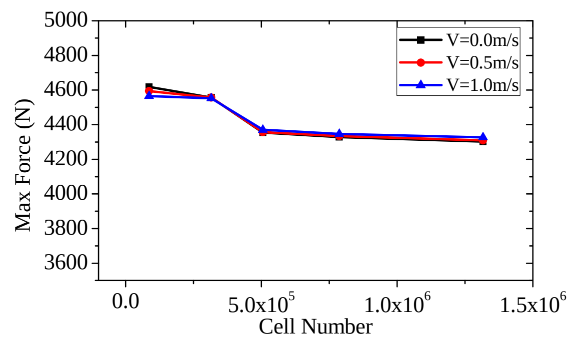


Fig. 7. Grid Convergence of Max Force.

Time Step Convergence Study

In OpenFOAM, adaptive time step control is available with the variable max Courant Number (MaxCo), as Eq. 8 shows. In the study of time step convergence, four max Courant Numbers were chosen: 0.50, 0.10, 0.20 and 0.30. Usually, the bigger the Courant Number, the larger the time step is. Three initial velocities were also considered for each value of the max Courant Number. Similarly, the validation objects were also the impact force f_I and max force f_M . We used Medium mesh (Mesh III) to study of time step convergence.

Fig. 8 is the result of impact force. It can be seen that, for the largest time step where MaxCo=0.3, the results of impact force were relatively underpredicted. However, as MaxCo decreased, the results of impact force became closer, which showed that smaller time step could capture the impact force better than a bigger one. It was reasonable because a big time step may jump over the peak value of the sharply increasing forces, while small time step is more likely to capture the peak value. Besides, the results also showed good time step convergence for the difference in results became smaller as MaxCo decreased.

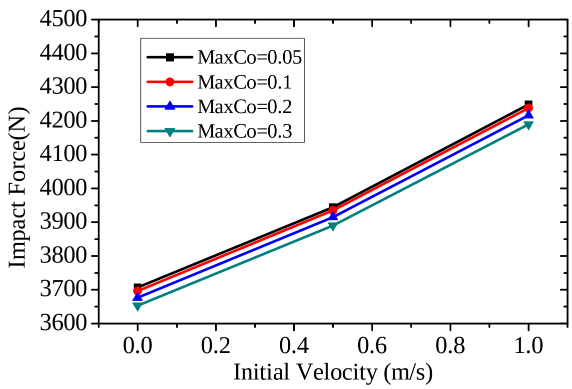


Fig. 8. Time Step Convergence of Impact Force

Fig. 9 shows the results of max force, where we can see that the max force is less susceptible to the different values time step. The difference in max force between the maximum time step and minimum one was less than 10N (less than 0.2% of the average max force), while for the impact force, it was about 50N (about 1.5% of the average impact force). That is because max force was primarily composed of buoyance and resistance force without the impact effect and max force grew much more slowly than the impact force. This is the major reason why time step has more influence on the results of impact force.

Although the differences among the results of max force were small, the results indicated good grid-convergence of the prediction. For the case of maximum time step (MaxCo=0.3), the results were overpredicted, while for the MaxCo=0.05 and MaxCo=0.1, the results were quite close. The difference became smaller with the decrease of MaxCo. This showed good time step convergence.

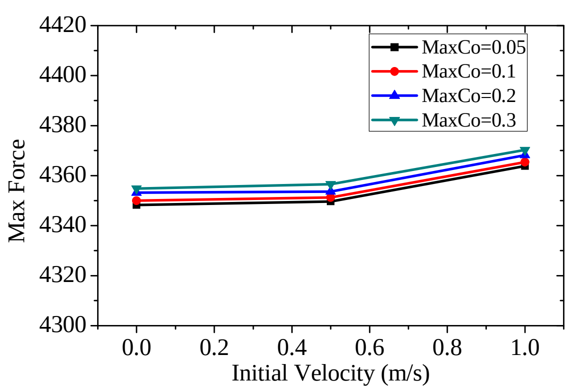
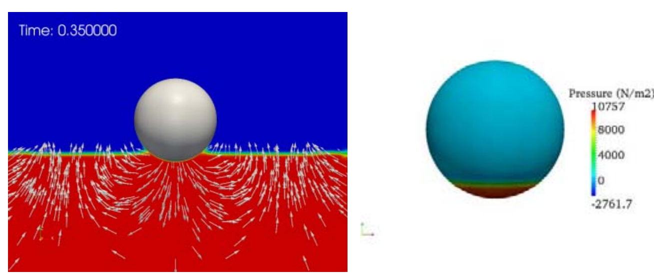


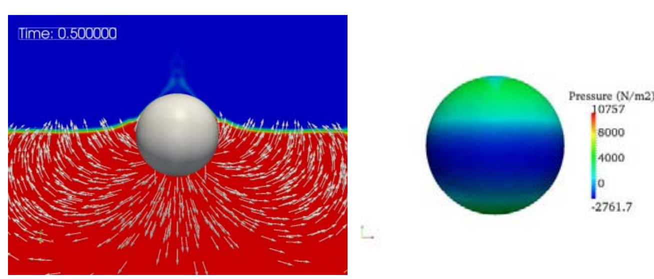
Fig. 9. Time Step Convergence of Max Force

Viscous Flow and Pressure Distribution

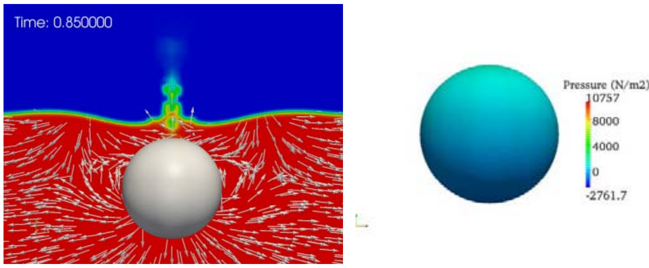
In this part, we focus on the velocity field and pressure distribution of the sphere. Here, we take the case of initial velocity equal to zero and the finest mesh (Mesh V) for example. Since the sphere is axial-symmetric, we clip a transverse of the computational domain, as the left column of Fig. 10. The contours were colored by volume of fraction. The red color indicated fluid of water and the blue side represented air. We used vectors to illustrate the velocity of the field. In order to show the vectors clearly, the velocity of air was neglected during the post-processing. For the right column of Fig. 10, the pressure contours were presented to illustrate the surface pressure distribution of the sphere. The 'pressure' here indicates dynamic pressure, excluding static pressure from the total pressure. At t=0.35s, sphere just impacted on the surface. It caused the flow right below the sphere moving down and the flow near the two sides of sphere going up, which resulted in an axial symmetry rotation of flow in the water. For the pressure distribution, it can be seen that extremely high pressure existed at the bottom region because of impact force. The maximum pressure reached 10757 N/m^2, while for the part of sphere in the air, pressure was nearly zero. This kind of sharp increase of pressure may cause great damage to structure. At t=0.5s, most part of the sphere entered into the water. At that moment, the rotation of velocity increased. The pressure was negative in the mid-region of the sphere, while for the up and bottom region, it was relatively large. At t=0.85s, the sphere was fully immersed. At this time, the sphere slowed down. The direction of the velocity changed from up-and-down to outward. The pressure distribution was relatively uniform for the velocity was small the moment when sphere touched the surface. At t=1.5s, the sphere began returning to the air. Compared with the results of t=0.5s, the direction of velocity is completely reversed. The pressure distribution was similar with that of t=0.5s. However, the pressure on sphere's top was slightly larger than other region. At t=1.7s, the sphere was almost inside air but with a little of connection with the water. The flow went up but the flow in the 'gap' between sphere and water went down. The pressure was larger than before. This is reasonable because the surface tension, which created a large pressure force, prevented the sphere breaking thought the surface to the air. This phenomenon was quite similar with that when the sphere impacted on the surface from the air.



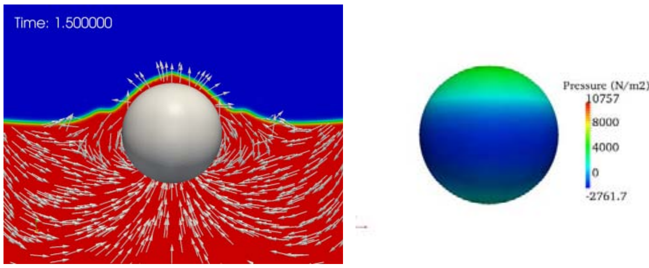
(a) Impacting Moment (t=0.35s)



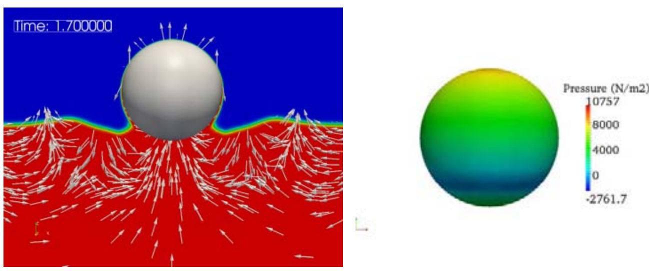
(b) Entering Water (t=0.5s)



(b) Fully Immersed (t=0.85s)



(c) Float up (t=1.5s)



(d) Jump to Air (t=1.7s)

Fig. 10. Velocity Vector (Left Column) and Pressure Contour (Right Column)

Effect of Sponge Layer

In order to check the effect of sponge layer, which was set up at the four sides of the tank, two additional cases were considered. One was with sponge layer setup and the other was not. Both cases were set up with zero initial velocity and one meter's height. We compared the time histories of sphere's displacement. For time ranging from 0s to 5s, the results were almost the same. However, the difference occurred after t=5s as in Fig. 11. For the case with sponge layer, the amplitude of the motion was continuously decreasing until reaching an equilibrium condition. However, for the case without sponge layer, the motion was quite different. The amplitude of motion increased instead of reduction. This was due to the waves, which were generated by the sphere and were reflected by the walls of the tanks. The reflected waves seriously disturbed the motion of sphere. Therefore, the sponge layer took effect and absorbed the waves induced by the impact. In order to reach better function of the sponge layer, sometimes, the length of the layer need to be increased.

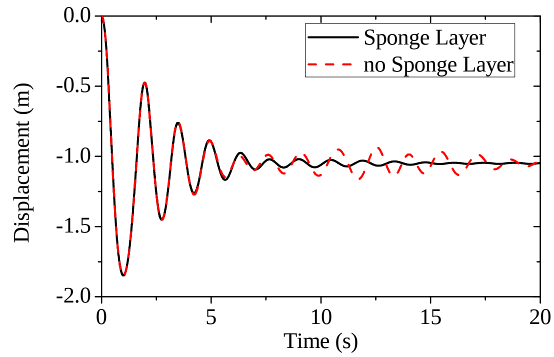


Fig. 11 Time Histories of Motion Displacement of the Two Cases

CONCLUSION

Dynamic mesh method and VOF method in OpenFOAM are presented here to solve the sphere water entry problem. Grid and time step convergence study were concerned to exam the correctness and robustness of these methods. We have considered three different initial velocities in each case. The results showed that the procedure of sphere's down-and-up movement was well simulated. In the convergence study, the impact force and max force were captured accurately if the mesh size was larger enough and time step was relatively small. The results also indicated good grid and time step convergence. From the velocity flow, it can be seen that sphere caused flow rotation when entering the water. And the direction of rotation changed according to the motion direction of the sphere. The results of pressure distribution on the surface of showed that when sphere impacted on the free-surface, the surface tension caused large pressure concentration and sharp increase pressure at the bottom region. The similar thing happened in the moment when sphere break through the surface and returned to the air, but the pressure was relatively smaller than when entering water.

Although in present work, only one sphere was concerned, since the OpenFOAM code is easy to extend, it is not difficult to calculate multi-sphere problem in the future work. In the multi-sphere problem, the interaction between each sphere should be considered, which means a more complex motion of the spheres

However, there are still some problems to resolve. Although dynamic mesh is popular and easy to apply in OpenFOAM, the amplitude of motion is restricted due to the limitation of movement of mesh nodes. Therefore, how to develop a new method such as overset grid, fictitious boundary and immersed boundary, which can make the object move inside the computational domain arbitrarily without restriction is the major concern in our future work.

ACKNOWLEDGEMENTS

The support of National Natural Science Foundation of China (Grant No. 11072154, 50739004), Research Foundation of State Key Laboratory of Ocean Engineering of China (Grant No. GKZD010064) and the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning for this work is gratefully acknowledged.

REFERENCE

Battistin, D and Iafrati, A (2003). "Hydrodynamic loads during water entry of two-dimensional and axisymmetric bodies." *Journal of Fluids and Structures* Vol 17, No 5, pp 643-664.

Carrica, PM, Paik, KJ, et al. (2008). "URANS analysis of a broaching event in irregular quartering seas." *Journal of Marine Science and Technology* Vol 13, No 4, pp 395-407.

Carrica, PM, Wilson, RV, et al. (2007). "Ship motions using single-phase level set with dynamic overset grids." *Computers and Fluids* Vol 36, No 9, pp 1415-1433.

Hirt, CW and Nichols, BD (1981). "Volume of fluid (VOF) method for the dynamics of free boundaries." *Journal of Computational Physics* Vol 39, No 1, pp 201-225.

Iafrati, A and Korobkin, AA (2008). "Hydrodynamic loads during early stage of flat plate impact onto water surface." *Physics of Fluids* Vol 20, No, pp 082104.

Issa, R (1986). "Solution of the implicitly discretised fluid flow equations by operator-splitting." *Journal of computational physics* Vol 62, No 1, pp 40-65.

Jasak, H (1996) "Error analysis and estimation for the finite volume method with applications to fluid flows" PhD Thesis, Imperial College

Jasak, H (2009). Dynamic mesh handling in OpenFOAM. *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Orlando, FLorida.

Jasak, H and Tukovic, Z (2006). "Automatic mesh motion for the unstructured finite volume method." *Transactions of FAMENA* Vol 30, No 2, pp 1-20.

Kleefsman, K, Fekken, G, et al. (2005). "A volume-of-fluid based simulation method for wave impact problems." *Journal of Computational Physics* Vol 206, No 1, pp 363-393.

Maruzewski, P, Le Touzé, D, et al. (2010). "SPH high-performance computing simulations of rigid solids impacting the free-surface of water." *Journal of Hydraulic Research* Vol 48, No SUPPL. 1, pp 126-134.

Osher, S and Sethian, JA (1988). "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations." *Journal of Computational Physics* Vol 79, No 1, pp 12-49.

Rhie, CM and Chow, WL (1983). "NUMERICAL STUDY OF THE TURBULENT FLOW PAST AN AIRFOIL WITH TRAILING EDGE SEPARATION." *Aiaa Journal* Vol 21, No 11, pp 1525-1532.

Rusche, H (2003) "Computational fluid dynamics of dispersed two-phase flows at high phase fractions" PhD Thesis, Imperial College

Wan, DC and Turek, S (2006). "Direct numerical simulation of particulate flow via multigrid FEM techniques and the fictitious boundary method." *International Journal for Numerical Methods in Fluids* Vol 51, No 5, pp 531-566.

Wan, DC and Turek, S (2007). "Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows." *Journal of Computational Physics* Vol 222, No 1, pp 28-56.