# A LATTICE BOLTZMANN APPROACH FOR THREE-DIMENSIONAL TSUNAMI SIMULATION BASED ON THE PLIC-VOF METHOD

Kenta Sato[1], Shunichi Koshimura[2]

Free surface flow problems occur in numerous disaster simulations, such as tsunamis inland penetration in urban area. Simulation models for these problems have to be non-hydrostatic and three-dimensional because of the strong non-linearity and higher-order physical phenomena. Despite all the progress in the modern computational fluid dynamics, such simulations still present formidable challenges both from numerical and computational cost point of view. The lattice Boltzmann method (LBM) has been attracting attention as an alternative fluid simulation tool to overcome the problems. In current study, LBM for three-dimensional tsunami simulations is developed which are coupled with the piecewise linear interface calculation with the Volume of Fluid (VOF) approach. This model is for an efficient three-dimensional tsunami simulation by a one-fluid formulation, where the lattice Boltzmann equation is assigned to solve for a single virtual fluid. Various benchmark problems are also carried out to validate the utility of the proposed models in term of coastal engineering.

*Keywords: the lattice Boltzmann method; the piecewise linear interface calculation; the volume of fluid method*

## INTRODUCTION

Three-dimensional free surface modelling is required to simulate tsunami inundation in urban areas with high accuracy. Despite offering a powerful tool to investigate tsunamis, such as grid-based (Choi *et al.*, 2007; Hu *et al.*, 2012; Wroniszewski *et al.*, 2014; Wu *et al.*, 2014) or meshfree methods (Koshizuka and Oka, 1996; Khayyer and Gotoh, 2009; Asai *et al.*, 2012; Sun *et al.*, 2015), three-dimensional tsunami simulations have been successfully applied only in limited computing environment. The overall implicit scheme of most Navier-Stokes solvers has to solve one or more linear system of equations in each time step (i.e., the pressure Poisson equation), this implies that the capillary effects could acts as a formidable computational barrier.

The lattice Boltzmann method (LBM) has been attracted attention as a novel computational fluid dynamics. LBM originates from the lattice gas automata for kinematic molecular dynamics (McNamara and Zanetti, 1988). In contrast to other solvers, LBM can be considered an alternative approach to solve macroscopic fluid equations. The key features of the LBM are as follows:

1. Fully explicit method in time integration, which means LBM does not have to solve the pressure Poisson equation, iteratively
2. Easy implementation with high-performance computing (Janßen and Krafczyk, 2011; Obrecht *et al.*, 2011; Rinaldi *et al.*, 2012; Obrecht *et al.*, 2013; Safi and Turek, 2016) due to the algorithmic operations and data locality

In this regard, we believe LBM has advantages as an alternative approach to perform three-dimensional tsunami simulations efficiently. In current study, we develop a three-dimensional free surface model by LBM based on the piecewise linear interface calculation (PLIC) method to enhance the model accuracy and overcome the traditional model problems (Thürey and Rüde, 2009). For the validation of the free surface model, we present simulations of dam breaking and breaking waves benchmarks.

## THE LATTICE BOLTZMANN METHOD

LBM evolved from methods for the simulation of gases that computed the motion of each molecule in the gas purely integer operations. In Hardy *et al.* (1976), there was a first attempt to perform fluid simulations with this approach. Motivated by this improvement (McNamara and Zanetti, 1988), they developed the algorithm that was actually called LBM by performing simulations with averaged floating point values instead of single fluid molecules. The important contribution to the basic LBM was the simplified collision operator with a single relaxation parameter. This collision model is well-known as the Bhatnagar-Gross-Krook (BGK) approximation (Bhatnagar *et al.*, 1954), and was derived independently by Chen *et al.* (1992) and Qian *et al.* (1992).

### The governing equation

The governing equation of LBM consists of two steps, the streaming and the collision step. In this study, a simple lattice BGK model was implemented for collision step. The lattice BGK equation, using a single relaxation time $\tau$ is given as:

$$f_i\left(\boldsymbol{x}+\boldsymbol{e}_i\Delta t, t+\Delta t\right) = f_i\left(\boldsymbol{x}, t\right) - \frac{1}{\tau}\left[f_i\left(\boldsymbol{x}, t\right) - f_i^{eq}\left(\rho, \boldsymbol{u}\right)\right] \tag{1}$$

[1] Graduate School of Engineering, Tohoku University, Japan

[2] International Research of Disaster Science, Tohoku University, Japan

in which, $\rho$ and $\boldsymbol{u}$ are the macroscopic fluid density and velocity, given by the 0th and 1st order moments of the distribution function $f_i$:

$$\rho = \sum_i f_i \tag{2}$$

$$\rho_0 \boldsymbol{u} = \sum_i \boldsymbol{e}_i f_i \tag{3}$$

and $\tau$ determines the macroscopic kinematic fluid viscosity $\nu$ as:

$$\nu = \left(\tau - \frac{1}{2}\right)\frac{\Delta x^2}{\Delta t} \tag{4}$$

where $\Delta x$ and $\Delta t$ are the spacing lattice size and time step interval, respectively. For each of the lattice velocities $\boldsymbol{e}_i$, a floating point number $f_{0,\ldots,q-1}$, representing the fraction of particles moving with this velocity, needs to be stored. The virtual particle movement is restricted into a limited number of directions. In this thesis, We used the three-dimensional 19-speed (D3Q19) square lattice model for three-dimensional fluid simulation. Alternatives are models with 15 or 27 velocities. However, the latter one has no apparent advantages over the 19 velocity model, while the model with 15 velocities has a decreased stability. The D3Q19 model is, thus, usually preferable as it requires less memory than the 27 velocity model. The D3Q19 model with its lattice velocity vectors $\boldsymbol{e}_{0,\ldots,18}$ is shown in Fig. 1 The velocity vectors take the following values:

$$\boldsymbol{e}_i = \begin{bmatrix} 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \end{bmatrix} \tag{5}$$

In the D3Q19 model there are particles not moving at all ($f_0$), moving with speed 1 ($f_{1,\ldots,6}$) and moving with speed $\sqrt{2}$ ($f_{7,\ldots,18}$). In the following, a subscript of $I$ will denote the value from the inverse direction of a value with subscript $i$. $f_i$ and $f_I$ are, thus, opposite distribution functions with inverse velocity vectors $\boldsymbol{e}_I = -\boldsymbol{e}_i$.

#### The equilibrium distribution function

The continuity equation in LBM introduces a compressibility effect which become critical for the simulation of the incompressible flows (Chen *et al.*, 1992). In order to reduce the compressibility effect for the time dependent flows, we need the following incompressible LBM (He and Luo, 1997), a slight modification of the compressible one with respect to the density.

It is well understood that in incompressible flows the density is a constant, say $\rho_0$, and the density fluctuation $\delta\rho$, should be of the order $O\left(\mathrm{Ma}^2\right)$ in the limit of Ma → 0. If we explicitly substitute $\rho = \rho_0 + \delta\rho$ into the equilibrium distribution function (Qian *et al.*, 1992) and neglect the terms proportional to $\delta\rho\left(\boldsymbol{u}/c\right)$ and $\delta\rho\left(\boldsymbol{u}/c\right)^2$, which are of the order $O\left(\mathrm{Ma}^3\right)$ or higher, then the equilibrium distribution function becomes:

$$f_i^{eq}(\rho, \boldsymbol{u}) = w_i\left[\rho + \rho_0\left(\frac{(\boldsymbol{e}_i \cdot \boldsymbol{u})}{c_s^2} + \frac{(\boldsymbol{e}_i \cdot \boldsymbol{u})^2}{2c_s^4} - \frac{(\boldsymbol{u} \cdot \boldsymbol{u})}{2c_s^2}\right)\right] \tag{6}$$

where $c_s$ is the virtual sound speed, $w_i$ is the weight coefficient given as;

$$w_i = \begin{cases} 1/3 & (i = 1) \\ 1/18 & (i = 2, \cdots, 7) \\ 1/36 & (i = 8, \cdots, 19) \end{cases} \tag{7}$$

for D3Q19 square lattice model.

The above equation is the equilibrium distribution function of the incompressible LBM. By the same Chapman-Enskog expansion procedure, we can obtain the incompressible Navier-Stokes equations accurate up to the second order of Mach number $O\left(\mathrm{Ma}^2\right)$ for continuity equation and third order of Mach number $O\left(\mathrm{Ma}^3\right)$ for momentum equations as:

$$\nabla \cdot \boldsymbol{u} = 0 + O\left(\mathrm{Ma}^2\right) \tag{8}$$

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\,\boldsymbol{u} = -\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{g} + O\left(\mathrm{Ma}^3\right) \tag{9}$$

Notice that the above system is still biased from standard Navier-Stokes equations by the compressibility effect related to Mach number. In order to reduce this compressibility effect, small Mach number is preferable, which means that $\Delta t$ has to be small with respect to $\Delta x$.

### Solid wall boundary condition

In LBM, there is no explicit distribution function $f_i$ streaming from the boundary into the bulk, we have to construct the unknown distribution functions. For no-slip solid wall boundaries, we imposed a second-order bounce-back scheme (Bouzidi *et al.*, 2001). It slightly violates the conservation of mass, but because it is second order in space, an overall scheme of second order can be achieved. The modified bounce-back scheme is given by:

$$f_I(\boldsymbol{x}, t + \Delta t) = f_i(\boldsymbol{x}, t) \tag{10}$$

where $f_I$ and $f_i$ are anti-parallel incoming (missing) and outgoing distribution functions, respectively.

### Interface boundary condition

When we apply the single-phase free surface model into LBM, another problem be occurred in the interface cells. The distribution functions of empty gas cells are never accessed. Interface cells, however, always have empty cell neighbours (Fig. 2). Therefore, during the streaming step only distribution functions from fluid cells or other interface cells are streamed normally, while the distribution functions that would be read from empty cells need to be reconstructed with corresponding free surface boundary conditions at the interface cell (Körner *et al.*, 2005; Thürey and Rüde, 2009). This boundary condition does not require additional constructs, such as ghost layers around the interface. They, thus, can be treated locally for each cell. An atmospheric pressure of $\rho_A = 1.0$ is used, as this is also the reference density and pressure of the fluid. Moreover, it is assumed that the viscosity of the fluid is significantly lower than that of the gas phase, while having a higher density. Hence, the gas follows the fluid motion at the interface. In terms of distribution functions, this means that if at $(\boldsymbol{x} + \Delta t\boldsymbol{e}_i)$ there is an empty cell:

$$f_I(\boldsymbol{x}, t + \Delta t) = f_i^{eq}(\rho_A, \boldsymbol{u}) + f_I^{eq}(\rho_A, \boldsymbol{u}) - f_i(\boldsymbol{x}, t) \tag{11}$$

where $\boldsymbol{u}$ is the velocity of the interface cell at position $\boldsymbol{x}$ and time $t$. The pressure of the atmosphere onto the fluid interface is introduced by using $\rho_A$ for the density of the equilibrium distribution functions. Applying Eq. 11 to all directions with empty neighbour cells would result in a full set of distribution functions for interface cells. This boundary condition, thus, can be used for a multiphase flow (Anderl *et al.*, 2014) taking proper density instead of density for atmosphere. However, to balance the forces on each side of the interface, the distribution functions coming from the direction of the interface normal $\boldsymbol{n}$ are also reconstructed. Thus, as shown in Fig. 2, if the distribution function $f_i$ would be streamed from an empty gas cell, or if $\boldsymbol{n} \cdot \boldsymbol{e}_I > 0$ holds, $f_i$ is reconstructed using Eq. 11.

## ONE-FLUID FREE SURFACE MODEL FOR TSUNAMI MODELLING

Free surface flow problems involve an air and a water phase, so that, technically, such flows can be classified as multi-phase flows. However, in tsunami simulation, the flow behaviour is dominated by the water phase, which means that it is not necessary to resolve the full aerodynamics inside the air phase. We, thus, developed a three-dimensional free surface model by one-fluid formulation. When the governing equations are solved on a fixed grid, using one set of equations for the whole flow field, the different fluids have to be identified in some way. This is generally done by using a colour function that takes different values in the different fluids.

### Competing cells status

The Volume of Fluid (VOF) method (Hirt and Nichols, 1981; Youngs, 1982; Rudman, 1997) is the oldest approach to advect a colour function and - after many improvements and innovations - continues to be widely used for three-dimensional tsunami simulations due to the conservation of fluid mass. We also adopted VOF approach as a free surface model in LBM. The colour function $c$, which means fluid fraction value for each grid, is introduced as a new macroscopic value (Thürey and Rüde, 2009). This function is in the range $[0, 1]$.

### The piecewise linear interface reconstruction

Hirt and Nichols (1981) proposed an interface reconstruction method, in which it is approximated by straight lines, parallel to the coordinate directions. To determine whether the interface should be horizontal or vertical, Hirt and Nichols (1981) have found the normal to the interface, using values of $c$ in the neighboring cells, and selected the orientation of the interface depending on whether the normal was more closely aligned with the horizontal or the vertical axis. This approach is the simplest VOF model and commonly used in three-dimensional tsunami simulation as of today. Rudman (1997), however, suggested that the Hirt and Nichols (1981) method is not significantly more accurate.

Although the method of Hirt and Nichols (1981) perhaps did not improve significantly on the simple linear interface calculation (SLIC) method (Noh and Woodward, 1976), it nevertheless suggested that the key to improving the behaviour of the advection scheme was the reconstruction of the interface in each cell, using the value of the colour function in each cell, along with the value in the neighbouring cells. In the piecewise linear interface calculation (PLIC) method introduced by Youngs (1982), the interface is approximated by a straight-line segment in each cell. Despite this approach, thus, little more complicated in comparison with SLIC method, PLIC method is continue to be improved even now. For example, Multi-interface advection and reconstruction solver (MARS) has been proposed by Kunugi and Kino (2005) as an advanced PLIC algorithm for multiphase fluid analysis.

In this study, we implemented PLIC method as an interface reconstruction for more accurate free surface simulation by LBM. The orientation of the line is determined by the normal to the interface, which is found by considering the value $c$ in both the cell under consideration and in the adjacent cells. Once the interface in each cell has been constructed, the fluxes from one cell to another are computed by geometric considerations. The result of the advection generally depends on the accuracy of the interface reconstruction; finding the normal accurately, therefore, becomes critical for PLIC methods. In three dimension, the interface shapes are approximated as a trapezoid as shown in Fig. 3, and the line segment function is given in each interface cell as follows:

$$\boldsymbol{n} \cdot \boldsymbol{x} = n_1 x_1 + n_2 x_2 + n_3 x_3 = \alpha \tag{12}$$

where $\boldsymbol{n}$ is the unit interface normal and $\alpha$ is the segment from the Cartesian origin. The calculation algorithms of the PLIC method, thus, are as follows:

1. Calculate the interface normal $\boldsymbol{n}$

2. Calculate the segment parameter $\alpha$

3. Calculate the mass flux across the cells and advection interface cells

**Interface normal**

Although various algorithms have been proposed to determine the interface normal $\boldsymbol{n}$, Parker and Youngs method (Pilliod and Puckett, 2004) is commonly used for its simplicity and accuracy. In this approach, the interface normal $\boldsymbol{n}$ is calculated by the gradient of the fluid fraction $c$ direct, as follows:

$$\boldsymbol{n} = -\frac{\nabla c}{\|\nabla c\|} \tag{13}$$

The gradient is discretized by the central difference scheme from the surrounding values:

$$\nabla c = \frac{1}{2\Delta x} \begin{pmatrix} \bar{c}_x (x+1, y, z) - \bar{c}_x (x-1, y, z) \\ \bar{c}_y (x, y+1, z) - \bar{c}_y (x, y-1, z) \\ \bar{c}_z (x, y, z+1) - \bar{c}_z (x, y, z-1) \end{pmatrix} \tag{14}$$

where $\bar{c}$ is the average fraction level of neighboring cells as:

$$\bar{c}_x (x, y, z) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} c(x, y+i, z+j) \cdot w_{i,j} \tag{15}$$

$$\bar{c}_y (x, y, z) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} c(x+i, y, z+j) \cdot w_{i,j} \tag{16}$$

$$\bar{c}_z (x, y, z) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} c(x+i, y+j, z) \cdot w_{i,j} \tag{17}$$

where $w_{i,j}$ is the weighting parameter. The following weights achieve the best accuracy.

$$w_{i,j} = \begin{cases} 4 & (i, j) = (0, 0) \\ 2 & (i, j) = (1, 0), (0, 1) \\ 1 & (i, j) = (\pm 1, \pm 1) \end{cases} \tag{18}$$

### Interface segment parameter

Various algorithms can be used to determine the segment parameter $\alpha$. All algorithms must produce the same result in this step when the interface normal $\boldsymbol{n}$ is known. We, thus, used the simplest analytical algorithms (Scardovelli and Zaleski, 2000), which does not need iterative algorithm.

First, we standardized the calculated interface normal $\boldsymbol{n}$ as $n_1 > 0, n_2 > 0, n_3 > 0$ and $n_1 + n_2 + n_3 = 1$. When the interface normals were negative values, we transformed the coordinates to the normal non-negative. Second, we calculated the inverse problem $\alpha = f'(c, \boldsymbol{n})$ by utilising the fact that the fluid fraction $c$ is a function of the interface normal $n$ and the segment parameter $\alpha$ as follows:

$$c(\alpha, \boldsymbol{n}) = \frac{1}{6n_1 n_2 n_3}\left[\alpha^3 - \sum_{i=1}^{3} F_3(\alpha - n_i \Delta x_i) + \sum_{i=1}^{3} F_3(\alpha - \alpha_{\max} + n_i \Delta x_i)\right] \tag{19}$$

We determined $\alpha$ by solving the cubic equations analytically, where $\alpha_{\max}$ is the summation of the standardized interface normal as:

$$\alpha_{\max} = n_1 + n_2 + n_3 \tag{20}$$

and $F$ is the Heaviside function as:

$$F_3(x) = \begin{cases} x^3 & (x > 0) \\ 0 & (x \leq 0) \end{cases} \tag{21}$$

### Advection of the interface cell

In VOF method, the following advection equation has to be solved by an proper discretisation scheme:

$$\frac{\partial c}{\partial t} + \boldsymbol{u} \cdot (\nabla c) = 0 \tag{22}$$

In this study, the Lagrangian-explicit method (Aulisa *et al.*, 2003, 2007) was adopted to discretise Eq. 22. Fig. 4 also shows a schematic illustration of the method. In this method, the surface, which is approximated by line ab, moves to line cd with the face velocity at the next time step. The blue area in Fig. 4 is the outgoing mass flux toward the right-hand cell in the next time step. As a result, the fluid fraction $c^{t+1}$ can be determined by the incoming and outgoing mass flux as follows:

$$c_i^{t+1} = VL_{i+1} + VC_i + VR_{i-1} \tag{23}$$

where $VL$ is the mass flux toward the left-hand cell, $VR$ is the mass flux toward the right-hand cell, and $VC$ is the remaining mass flux.

## VALIDATION

### A classic dam breaking benchmark

We simulated some dam breaking benchmarks. Fig. 5 shows the initial condition of the dam breaking flow (Martin and Moyce, 1952). In this benchmark, we validated the convergence behaviour of the model accuracy depending on the competing grid size, which is an important factor in determining the applicability of the model to other problems from an engineering perspective. The model accuracy was evaluated based on the dimensionless position of the surge front in the time evolution.

Tab. 1 shows the calculation parameters. In this benchmark, the dimensionless time $T$ and distance from the left side of the competing domain $Z$ were defined using the realistic $t\,(\mathrm{s})$, $z\,(\mathrm{m})$, and $a\,(\mathrm{m})$ shown in Fig. 5 and gravity $g\left(\mathrm{m/s^2}\right)$ as follows (Martin and Moyce, 1952):

$$T = t\sqrt{\frac{2g}{a}} \tag{24}$$

$$Z = \frac{z}{a} \tag{25}$$

The effect of gate opening was considered to shift the plots (Nichols and Hirt, 1971).

Fig. 6 shows the dimensionless time series position of the surge front. Another experimental data from (Koshizuka and Oka, 1996) were also plotted. The results of our model were in good agreement with the experimental data from the Case2 and Case3 simulations. Our model has a convergence of the calculation capability that depends on the resolution through the simulations. We found that our model achieves results that match the experimental data well, and we can resolve at least 80 computing cells per the characteristic length $a$ in the benchmark.

On the other hand, the underestimations in the Case1 simulation are due to the difference in physical properties caused by the coarse grid size. The Reynolds number in the flow field will be smaller in the LBM if we cannot ensure a sufficient spacing resolution. To overcome this problem, it can be considered that another parameterisation approach for the relaxation time $\tau$, determined by not the viscosity $\nu$ but the Reynolds number in the flow. This method might fix the Reynolds number as turbulence. We, however, believe that a practical solution for the problem is to ensure high resolution in space because we could not determine the Reynolds number in unsteady flows, such a free surface flow. The solid wall boundary condition should also be treated carefully. Nishi and Doan (2013) have pointed out that three is room improvement for the modified bounce-back scheme especially in free surface model because the accuracy of the bounce-back scheme in the LBM depends on the relaxation time $\tau$.

### A dam breaking flow with a single obstacle

From the above section, the simple dam breaking benchmark was successfully simulated by our proposed free surface model. Here, this benchmark is extended with the addition of a solid obstacle in the centre of the flow domain by Kölke (2005) (Fig. 7).

Upon removal of gate, the collapsing of the water column is initiated and the front evolves and finally impacts the solid obstacle. At the moment, the flow pattern changes drastically, similar to what occurs during wave impact on structures. As the water height decreases, highly curved jet evolves, and the water is deflected to the top of the obstacle.

Fig. 8 compares between the calculated results with the experimental data by Kölke (2005). We fixed $\Delta x = 1.25 \times 10^{-3}$ and $\Delta t = 3.12 \times 10^{-6}$ to reduce the weak-compressibility. Very good agreement can be seen, at least for this qualitative comparison. The jet evolves slightly faster in computations tan in the experiment. Hence, we conclude that a proper grid resolution is essential, to capture at least the large-scale sprays and the resulting energy dissipation, and to accurately represent the solid body in conjunction with the above discussions. We also validated that, in LBM, a careful treatment of the corner boundaries is crucial to achieve good results. For example modified slip boundary conditions in LBM, an accurate normal vector is required in the corner cells, which is discontinuous points. If a smoothed geometry (Shigeto and Sakai, 2013; Sun and Sakai, 2015) is used, where the corner is rounded out and the corner node uses a linear interpolation of the neighbouring surface normal vectors, the jet shape is affected. For this reason, the normal vector at the obstacle corners is set to the normal vector of each respectively participating face. We should use a high-order scheme (Succi *et al.*, 1989; Maier *et al.*, 1996) for the boundary condition for a more precise simulation. In addition, some bubbles remained in the fluid domain and did not rise. Our model cannot address the bubble dynamics directly due to single-phase VOF approach. The remaining bubbles, however, negatively affected the stability of the model and the calculation accuracy of the interface normal $\boldsymbol{n}$. Numerical treatments are required to address the bubbles.

### Breaking wave in a rectangular tank

Finally, we simulated a three-dimensional breaking wave at an air-water interface. Numerical simulations based on fully non-linear potential flow theory (Longuet-Higgins and Cokelet, 1976; Grilli *et al.*, 1989) showed that a periodic sinusoidal wave of large amplitude, with initial velocities specified from linear wave theory, is not stable in a fully non-linear model and rapidly overturns and breaks.

The initial velocity field and interface shape for a two-dimensional linear wave of height $H$ in depth $h$, specified in a three-dimensional, are given by Lubin *et al.* (2006) with $z$ denoting the vertical direction and $z = 0$ at the undisturbed air-water interface:

$$\eta = \frac{H}{2} \cos(kx) \tag{26}$$

$$u_x = \frac{H}{2} \sigma \frac{\cosh k(h+z)}{\sinh(kh^*)} \cos(kx) \tag{27}$$

$$u_z = \frac{H}{2} \sigma \frac{\sinh k(h+z)}{\sinh(kh^*)} \sin(kx) \tag{28}$$

$$u_y = 0 \tag{29}$$

where $\sigma = 2\pi/T$ is the wave angular frequency, $k = 2\pi/L$ is the wave number, and $h^* = h + \epsilon$, where $\epsilon(y) = 0.08y$ is a small linear perturbation of the seafloor geometry. In this thesis, we set $L = 0.1$ (m) and $T = 0.308$ (s). As a result, the initial velocity profiles is shown in Fig. 9.

Fig. 10 compares the calculated interface shapes between our results and the reference data (Lubin *et al.*, 2006). We fixed the spacing resolution as $(x, y, z) = (1024, 128, 256)$ in this simulation.

As expected, the wave quickly overturns and develops a plunging jet. The agreement of our results with Lubin *et al.* (2006) results is good, although the shape after breaking wave was a little weakened. This is caused by the averaging extrapolation approach for empty cells, which become interface cells at the next time step. The extrapolation velocities for the empty cell crucial affects a numerical accuracy in the VOF approach. However, using high order schemes easily causes a numerical instability. We, thus, used averaging extrapolation approach in this study.

## CONCLUSION

In this study, we presented the implementation the single-phase free surface model based on PLIC approach to LBM to enable three-dimensional tsunami modellings more efficient. Moreover, we performed the validation of the proposed model in classic dam breaking benchmark and breaking waves problems. The following conclusions were drawn from the discussions above:

- We have developed a fully explicit three-dimensional free-surface model by LBM. Our model reduces interface oscillations and diffusions by PLIC approach.
- We found that the proposed model can overcome the defects of the conventional free surface model in LBM, and can simulate the flow field naturally.
- We determined the appropriate spacing resolution by the characteristic length of the dam breaking benchmarks. Moreover, we demonstrated the application to other problems.

LBM, however, might require finer resolution than that of other computational fluid dynamic. The adaptive mesh refinement (Zabelok *et al.*, 2015; Onodera and Idomura, 2018) technique will be needed to execute more efficient simulations. In addition to model development, we will investigate wave forces acting on structures (Kleefsman *et al.*, 2005) by our model, which allows the weak-compressibility to appear from the viewpoint of coastal engineering.

## ACKNOWLEDGEMENTS

This study was supported by KAKENHI (Grants-in-Aid for Scientific Research) 16K12845, 17J06615.

## References

Anderl, D., Bogner, S., Rauh, C., Rüde, U., and Delgado, A. Free surface lattice Boltzmann with enhanced bubble model. *Comput. Math. with Appl.*, 67(2):331–339, 2014.

Asai, M., Aly, A. M., Sonoda, Y., and Sakai, Y. A stabilized incompressible SPH method by relaxing the density invariance condition. *J. Appl. Math.*, 2012:1–24, 2012.

Aulisa, E., Manservisi, S., Scardovelli, R., and Zaleski, S. A geometrical area-preserving Volume-of-Fluid advection method. *J. Comput. Phys.*, 192(1):355–364, 2003.

Aulisa, E., Manservisi, S., Scardovelli, R., and Zaleski, S. Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry. *J. Comput. Phys.*, 225(2):2301–2319, 2007.

Bhatnagar, P. L., Gross, E. P., and Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Phys. Rev.*, 94(3):511–525, 1954.

Bouzidi, M., Firdaouss, M., and Lallemand, P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids*, 13(11):3452–3459, 2001.

Chen, H., Chen, S., and Matthaeus, W. H. Recovery of the Navier-Stokes equations using a lattice-gas Soltzmann method. *Phys. Rev. A*, 45(8):5339–5342, 1992.

Choi, B. H., Kim, D. C., Pelinovsky, E., and Woo, S. B. Three-dimensional simulation of tsunami run-up around conical island. *Coast. Eng.*, 54(8):618–629, 2007.

Grilli, S. T., Skourup, J., and Svendsen, I. A. An efficient boundary element method for nonlinear water waves. *Engng. Anal. Bound. Elem.*, 6(2):97–107, 1989.

Hardy, J., De Pazzis, O., and Pomeau, Y. Molecular dynamics of a classical lattice gas: Transport properties and time correlation functions. *Phys. Rev. A*, 13(5):1949–1961, 1976.

He, X. and Luo, L. S. Lattice boltzmann model for the incompressible Navier-Stokes equation. *J. Stat. Phys.*, 88(3/4):927–944, 1997.

Hirt, C. W. and Nichols, B. D. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.*, 39(1):201–225, 1981.

Hu, K. C., Hsiao, S. C., Hwung, H. H., and Wu, T. R. Three-dimensional numerical modeling of the interaction of dam-break waves and porous media. *Adv. Water Resour.*, 47:14–30, 2012.

Janßen, Christian and Krafczyk, Manfred. Free surface flow simulations on GPGPUs using the LBM. *Comput. Math. with Appl.*, 61(12):3549–3563, 2011.

Khayyer, A. and Gotoh, H. Modified moving particle semi-implicit methods for the prediction of 2D wave impact pressure. *Coast. Eng.*, 56(4):419–440, 2009.

Kleefsman, K. M. T., Fekken, G., Veldman, A. E. P., Iwanowski, B., and Buchner, B. A Volume-of-Fluid based simulation method for wave impact problems. *J. Comput. Phys.*, 206(1):363–393, 2005.

Kölke, A. *Modellierung und diskretisierung bewegter diskontinuitäten in randgekoppelten Mehrfeldsystemen*. PhD thesis, Technische Universität Braunschweig, 2005.

Körner, C., Thies, M., Hofmann, T., Thürey, N., and Rüde, U. Lattice Boltzmann model for free surface flow for modeling foaming. *J. Stat. Phys.*, 121(1-2):179–196, 2005.

Koshizuka, S. and Oka, Y. Moving-Particle Semi-implicit method for fragmentation of incompressible fluid. *Nucl. Sci. Eng.*, 123:421–434, 1996.

Kunugi, T. and Kino, C. DNS of falling film structure and heat transfer via MARS method. *Comput. Struct.*, 83(6-7):455–462, 2005.

Longuet-Higgins, M. S. and Cokelet, E. D. The deformation of steep surface waves on water: a numerical method of computation. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 350, pages 1–26. The Royal Society, 1976.

Lubin, P., Vincent, S., Abadie, S., and Caltagirone, J. P. Three-dimensional Large Eddy Simulation of air entrainment under plunging breaking waves. *Coast. Eng.*, 53(8):631–655, 2006.

Maier, R. S., Bernard, R. S., and Grunau, D. W. Boundary conditions for the lattice Boltzmann method. *Phys. Fluids*, 8(7):1788–1801, 1996.

Martin, J. C. and Moyce, W. J. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philos. Trans. R. Soc. A Math. Phys. Eng.*, 244:312–324, 1952.

McNamara, G. R. and Zanetti, G. Use of the boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.*, 61(20):2332–2335, 1988.

Nichols, B D and Hirt, C W. Improved free surface conditions for numerical incompressible flow computations. *J. Comput. Phys.*, 8:434–448, 1971. ISSN 00219991.

Nishi, Y. and Doan, P. V. Hybrid boundary condition combined with data assimilation for simulations of free surface flows using lattice Boltzmann method. *Comput. Fluids*, 88:108–114, 2013.

Noh, W. F. and Woodward, P. SLIC (simple line interface calculation). In *Proceedings, Fifth International Conference on Fluid Dynamics*, volume 59, pages 330–340, Berlin, 1976. Springer.

Obrecht, C., Kuznik, F., Tourancheau, B., and Roux, J. J. A new approach to the lattice Boltzmann method for graphics processing units. *Comput. Math. with Appl.*, 61(12):3628–3638, 2011.

Obrecht, C., Kuznik, F., Tourancheau, B., and Roux, J. J. Scalable lattice Boltzmann solvers for CUDA GPU clusters. *Parallel Comput.*, 39(6-7):259–270, 2013.

Onodera, N. and Idomura, Y. Acceleration of wind simulation using locally mesh-refined lattice boltzmann method on gpu-rich supercomputers. In *Supercomputing Frontiers*, pages 128–145, Cham, 2018. Springer International Publishing.

Pilliod, J. E. and Puckett, E. G. Second-order accurate volume-of-fluid algorithms for tracking material interfaces. *J. Comput. Phys.*, 199(2):465–502, 2004.

Qian, Y. H., D'Humières, D., and Lallemand, P. Lattice BGK models for Navier-Stokes Equation. *Europhys. Lett.*, 6(17):479–484, 1992.

Rinaldi, P. R., Dari, E. A., Vénere, M. J., and Clausse, A. A Lattice-Boltzmann solver for 3D fluid simulation on GPU. *Simul. Model. Pract. Theory*, 25:163–171, 2012.

Rudman, M. Volume-tracking methods for interfacial flow calculations. *Int. J. Numer. Methods Fluids*, 24:671–691, 1997.

Safi, M. A. and Turek, S. GPGPU-based rising bubble simulations using a MRT lattice Boltzmann method coupled with level set interface capturing. *Comput. Fluids*, 124:170–184, 2016.

Scardovelli, R. and Zaleski, S. Analytical relations connecting linear interfaces and volume fractions in rectangular grids. *J. Comput. Phys.*, 164(1):228–237, 2000.

Shigeto, Y. and Sakai, M. Arbitrary-shaped wall boundary modeling based on signed distance functions for granular flow simulations. *Chem. Eng. J.*, 231:464–476, 2013.

Succi, S., Foti, E., and Higuera, F. Three-dimensional flows in complex geometries with the lattice Boltzmann method. *Europhys. Lett.*, 10(5):433–438, 1989.

Sun, X. and Sakai, M. Three-dimensional simulation of gas-solid-liquid flows using the DEM-VOF method. *Chem. Eng. Sci.*, 134:531–548, 2015.

Sun, Z., Djidjeli, K., Xing, J. T., and Cheng, F. Modified MPS method for the 2D fluid structure interaction problem with free surface. *Comput. Fluids*, 122(February):47–65, 2015.

Thürey, N. and Rüde, U. Stable free surface flows with the lattice Boltzmann method on adaptively coarsened grids. *Comput. Vis. Sci.*, 12(5):247–263, 2009.

Wroniszewski, P. A., Verschaeve, J. C. G., and Pedersen, G. K. Benchmarking of Navier-Stokes codes for free surface simulations by means of a solitary wave. *Coast. Eng.*, 91:1–17, 2014.

Wu, Y. T., Yeh, C. L., and Hsiao, S. C. Three-dimensional numerical simulation on the interaction of solitary waves and porous breakwaters. *Coast. Eng.*, 85:12–29, 2014.

Youngs, D. Time-dependent multi-material flow with large fluid distortion. *Numer. Methods Fluid Dyn.*, (October):273–285, 1982.

Zabelok, S., Arslanbekov, R., and Kolobov, V. Adaptive kinetic-fluid solvers for heterogeneous computing architectures. *J. Comput. Phys.*, 303:455–469, 2015.

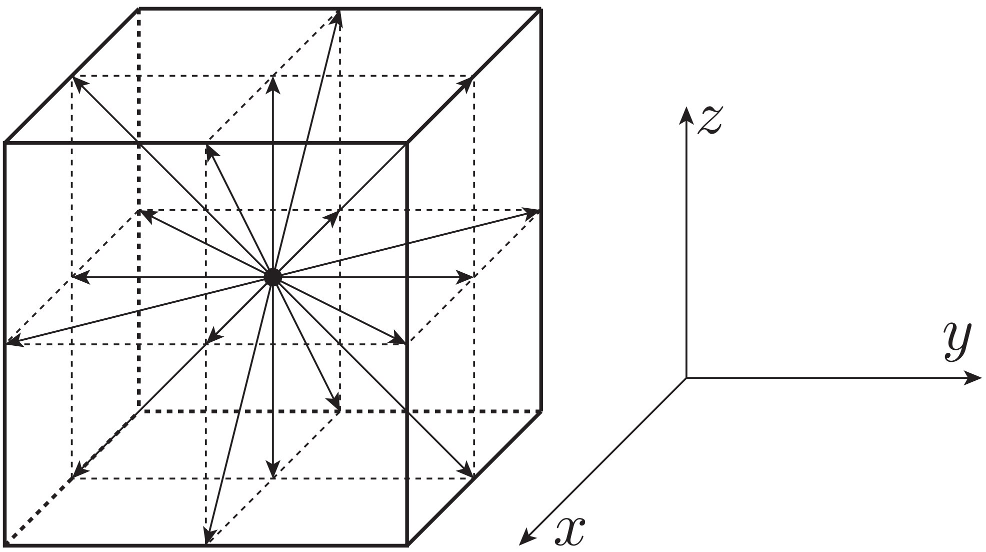

**Figure 1: Three-dimensional 19-speed (D3Q19) square lattice.**

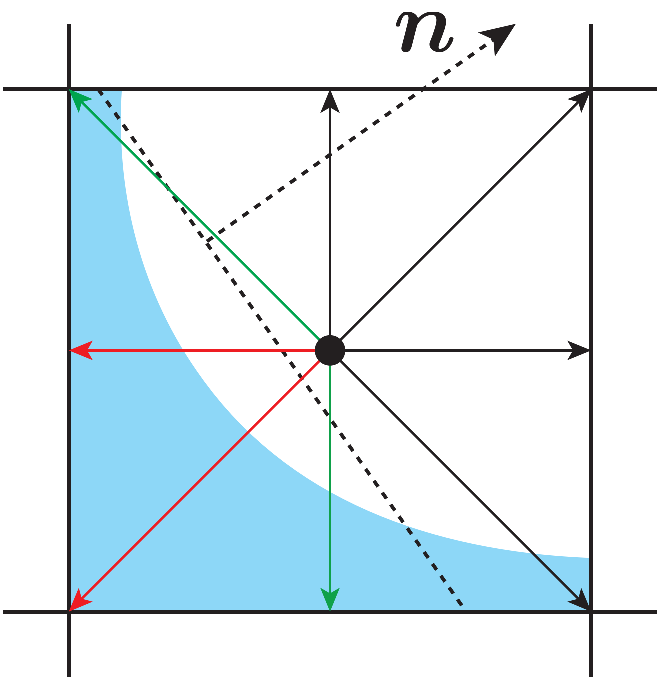

**Figure 2: The missing distribution functions $f_i$ on the interface cells, (red functions) functions from empty gas cells, (green functions) functions along the interface normal $n$ determined by VOF algorithms (broken line).**

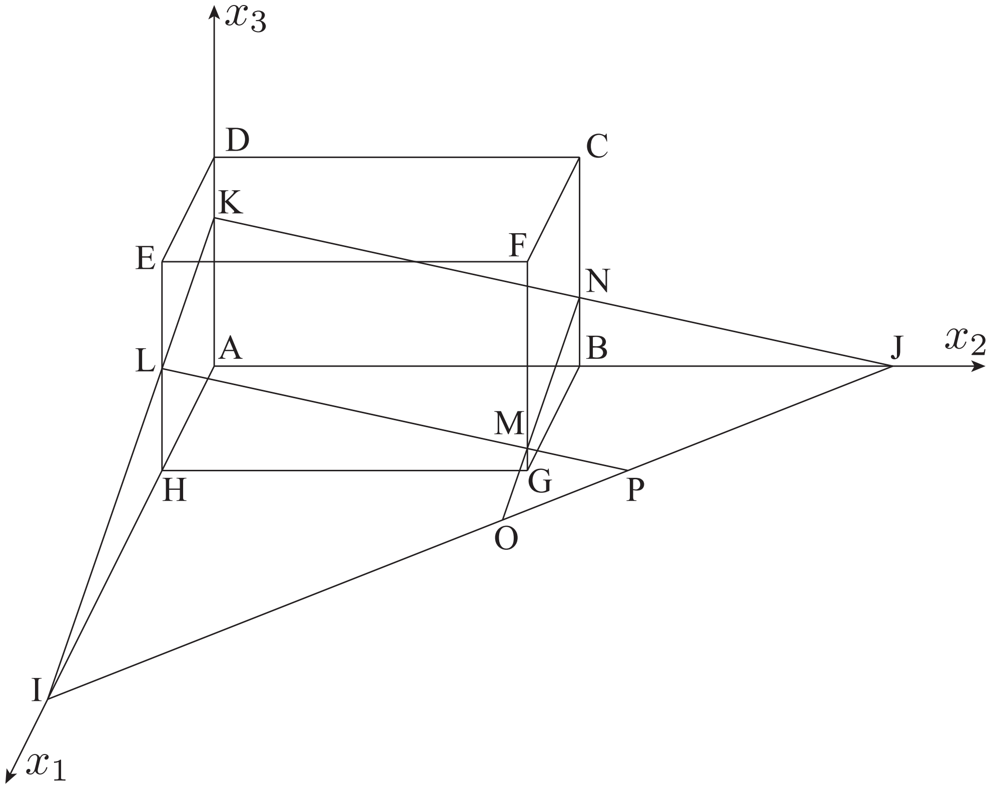

**Figure 3: A cut area is the region inside the parallelepiped ABCDEFGH and below the plane IJK.**

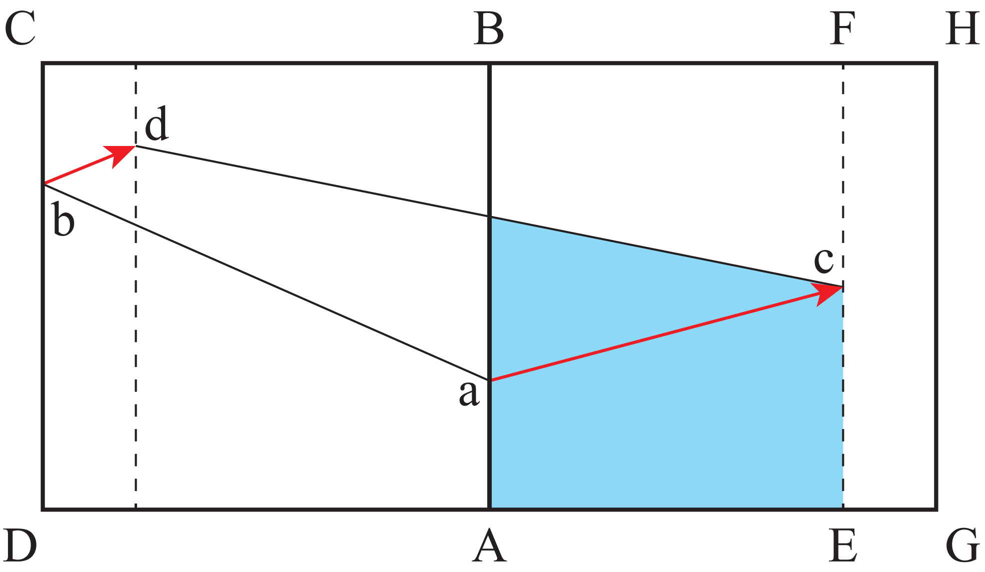

**Figure 4: A schematic illustration of the Lagrangian-explicit method in two dimensions. Cell ABCD is the upwind cell for cell AGHB. Line ab is advected to line cd. The volume gained is represented by the downwind blue area.**

**Table 1: Calculation parameters of the dam breaking flow (Martin and Moyce, 1952).**

| Case No. | Resolution | $\Delta x$ (m) | $\Delta t$ (s) | $\tau$ |
|---|---|---|---|---|
| Case1 | $200 \times 40 \times 60$ | $2.86 \times 10^{-3}$ | $7.14 \times 10^{-6}$ | $0.5 + 2.6 \times 10^{-6}$ |
| Case2 | $400 \times 80 \times 120$ | $1.43 \times 10^{-3}$ | $3.57 \times 10^{-6}$ | $0.5 + 5.3 \times 10^{-6}$ |
| Case3 | $800 \times 160 \times 240$ | $7.14 \times 10^{-4}$ | $1.79 \times 10^{-6}$ | $0.5 + 1.1 \times 10^{-5}$ |

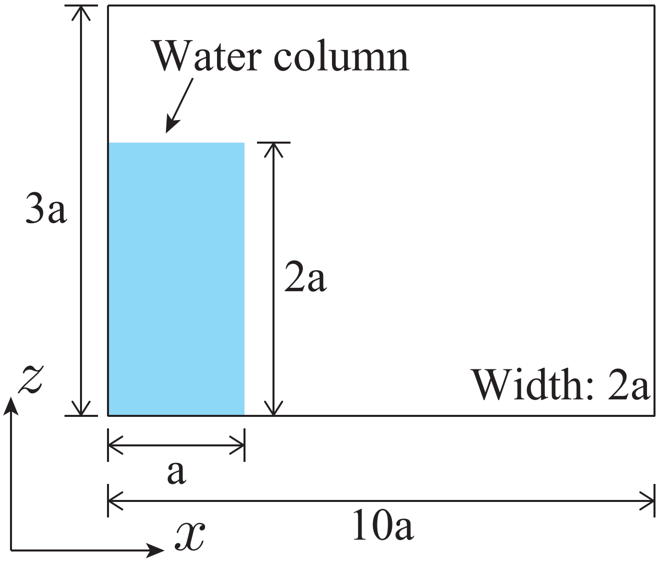

**Figure 5: The calculation domain of the Martin and Moyce (1952) dam breaking flows.**

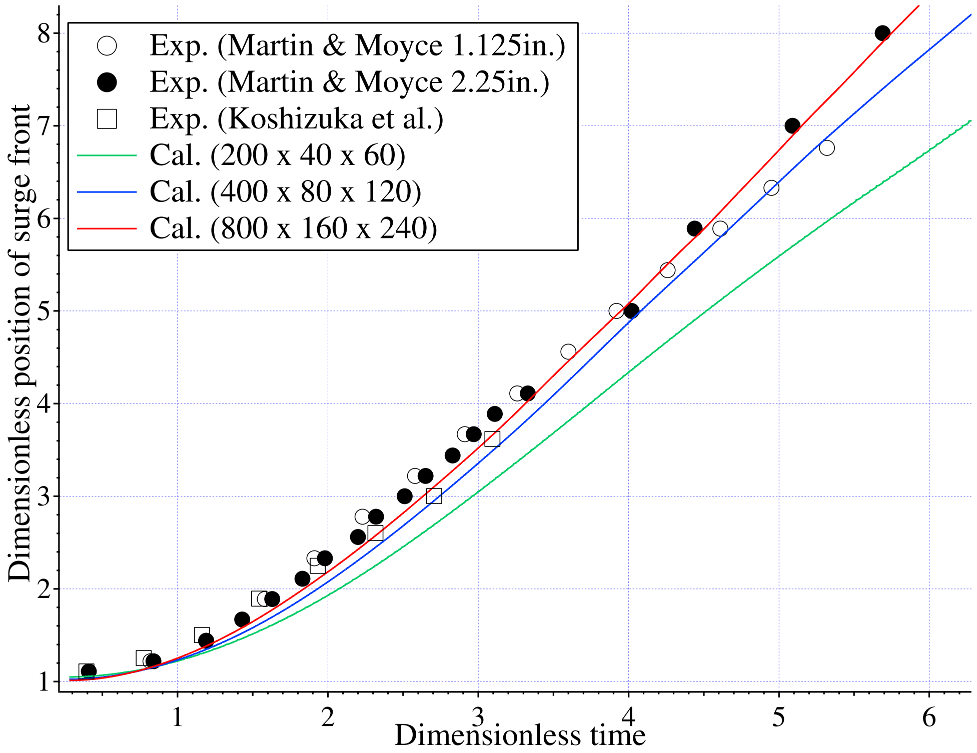

**Figure 6: Timeseries of the dimensionless position of the surge front.**

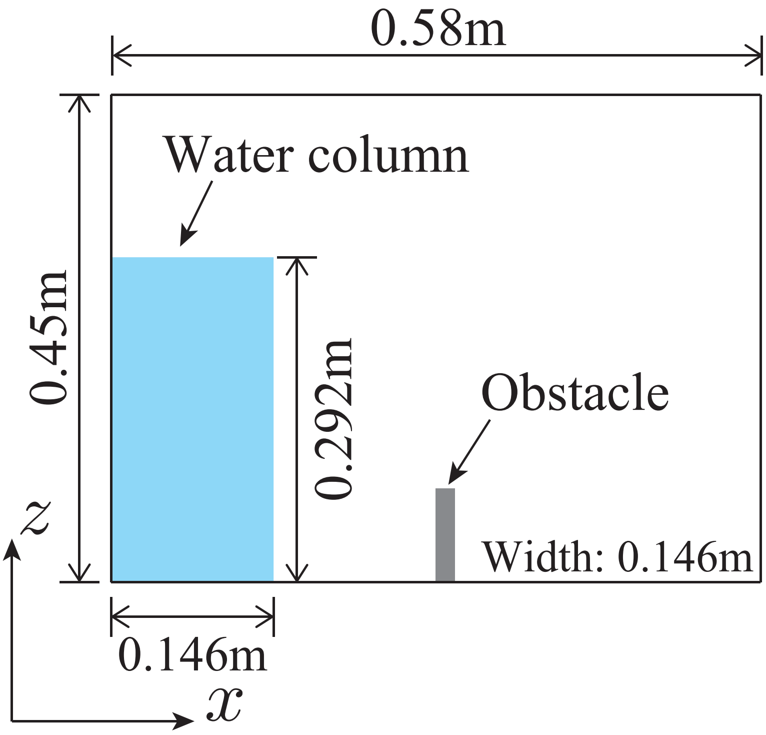

**Figure 7: The calculation domain of the Kölke (2005) dam breaking flows.**

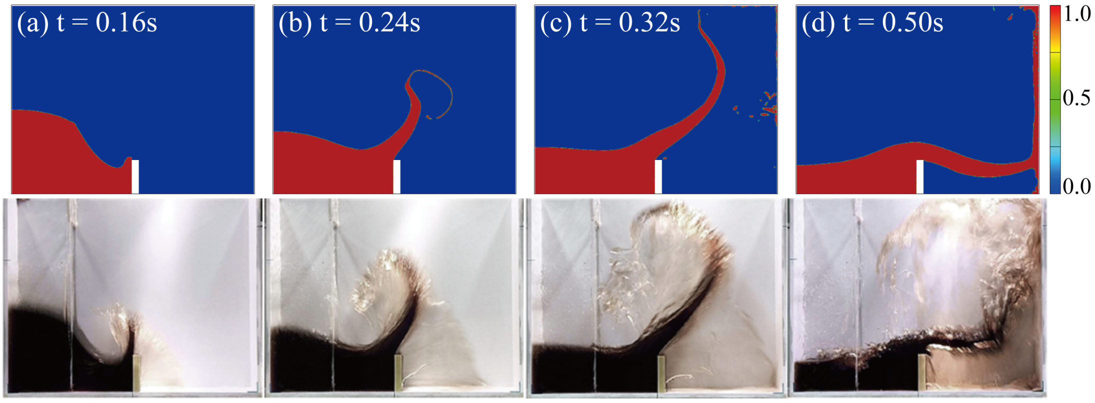

**Figure 8: The comparison of the interface shapes in two-dimensional view, (top) our results, (bottom) experimental data (Kölke, 2005).**

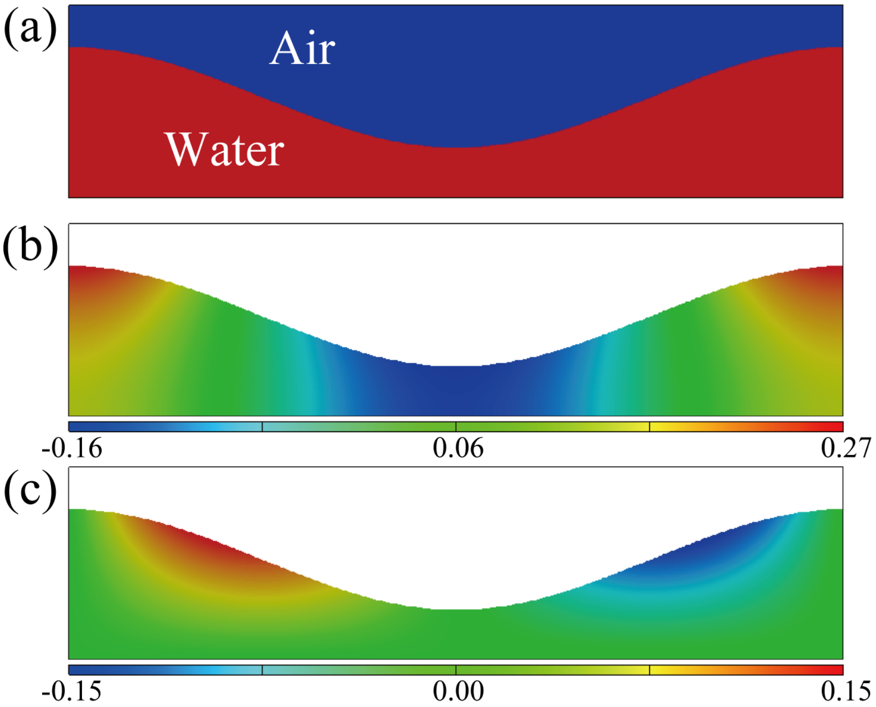

**Figure 9: Initial velocity profiles for three-dimensional breaking waves benchmark (Lubin *et al.*, 2006), (a) initial phase profile, (b) velocity along $x$-axis, (c) velocity along $y$-axis.**

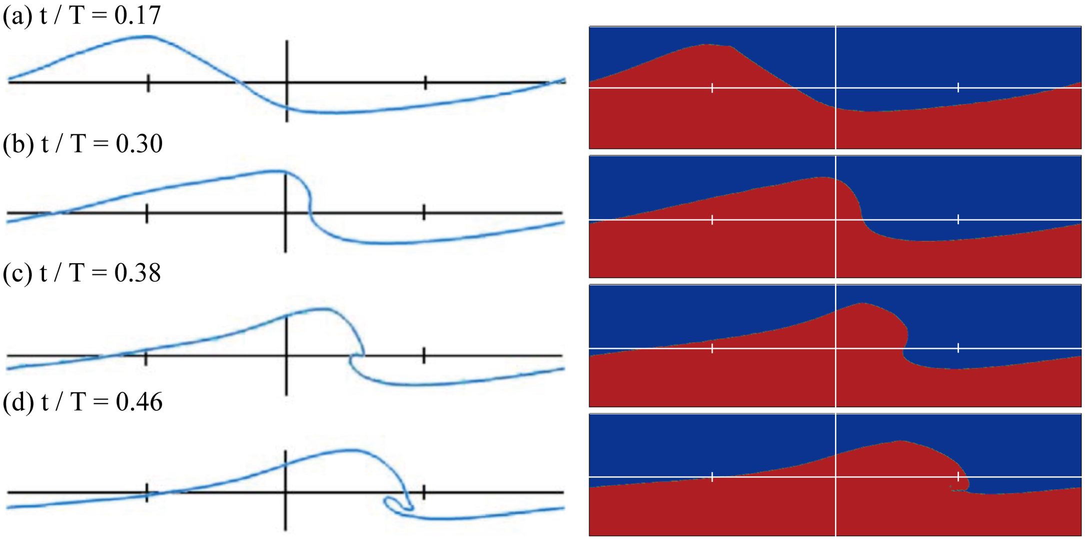

**Figure 10: Time evolution of the three-dimensional breaking wave. (left) simulation results (Lubin *et al.*, 2006), (right) our results.**