Proceedings of Computational Engineering Conference JSCES
Vol. 28, 2023 May

A High-Resolution Simulation for Driftwood Capture by Using AMR-LBM

Dawei Shen[1)], Takayuki Aoki[2)], Seiya Watanabe[3)] and Joji Shima[4)]

1) PhD student, Department of Mechanical Engineering, Tokyo Institute of Technology
(2-12-1 i7-3 O-okayama, Meguro-ku, Tokyo 152-8550, Japan, E-mail: shen@sim.gsic.titech.ac.jp)
2) Dr. Sci., Professor, Global Scientific Information and Computing Center, Tokyo Institute of Technology
(2-12-1 i7-3 O-okayama, Meguro-ku, Tokyo 152-8550, Japan, E-mail: taoki@gsic.titech.ac.jp)
3) Kyushu University (E-mail: swatanabe@riam.kyushu-u.ac.jp)
4) Sabo & Landslide Technical Center (E-mail: j-shima@stc.or.jp)

Flood disasters including driftwood pose a significant threat to human life and property safety. Thus, developing effective driftwood trapping devices installed on concrete dams is crucial. In this study, we conduct computational fluid dynamics (CFD) simulations of driftwood trapping experiments using the cumulant lattice Boltzmann method (LBM) for the fluid phase and the discrete element method (DEM) for the solid phase. To maintain mass conservation, we employ a conservative Allen-Cahn equation as an interface capturing method. We use CUDA programming and adaptive mesh refinement (AMR) to increase computational efficiency. Finally, we compare our simulation results with the experimental results.

Key Words : *CFD, Cumulant LBM, DEM, CUDA, AMR*

1. INTRODUCTION

In 2022, a severe flood in the Kyushu region of Japan caused serious casualties and property damage. The flood, which exerted a large fluid force, can cause significant structural damage when it accompanies floating objects such as driftwood [1].

To simulate this type of free surface flow, computational fluid dynamics is a useful tool, and the lattice Boltzmann method (LBM) has become an important alternative method in the field. LBM has several advantages, including its simple algorithm, suitability for massively parallel computations, and ease of implementation for complex geometries [2]. In this study, we utilize the cumulant LBM [3] to handle free surface flow with a high Reynolds number.

However, when using orthogonal grids for calculations, high resolution is rarely required for the entire computational domain. To achieve high-resolution calculations in specific areas where it is needed, we employ the adaptive mesh refinement method [4, 5], which reduces calculation costs and memory usage.

In this research, we utilize the lattice Boltzmann method with the adaptive mesh refinement method to simulate the driftwood trap devices experiment [6] carried out by Dr. Shima with high-resolution meshes, and present the results of our simulations in this paper.

2. NUMERICAL METHODS

(1) Cumulant Lattice Boltzmann Method

The lattice Boltzmann method (LBM) assumes that the fluid can be represented as a set of virtual particles that propagate and collide on lattice points. To solve for fluid flow, LBM requires solving the following equations:

$$f_{ijk}(\boldsymbol{x}+\boldsymbol{c}_{ijk}\Delta t, t+\Delta t) = f_{ijk}(\boldsymbol{x},t) - \frac{1}{\tau}\left[f_{ijk}(\boldsymbol{x},t) - f_{ijk}^{eq}(\boldsymbol{x},t)\right] + F_{ijk} \qquad (1)$$

where f is the velocity distribution function, $\boldsymbol{x} = (x, y, z)$ is the position, t is the time, and ijk denotes the direction of velocity distribution function. In this paper, we use the D3Q27 lattice model, so $(i, j, k) \in [-1,0,1]^3$. Here, $c = \Delta x/\Delta t$ is the lattice speed, with Δx and Δt being one lattice unit in a uniform grid. τ is relaxation time, f^{eq} is the Maxwell distribution function in discrete velocity space, and F is the external force.

To improve numerical stability, we adopt the cumulant collision model in this paper. The cumulant can be derived from the distribution function using the following equations:

$$C_{\alpha\beta\gamma} = C^{-\alpha-\beta-\gamma}\frac{\partial^\alpha\partial^\beta\partial^\gamma}{\partial\Xi^\alpha\partial\Upsilon^\beta\partial Z^\gamma}\ln(F(\Xi))\bigg|_{\Xi=0} \qquad (2)$$

$$F(\Xi) = \mathcal{L}[f(\xi-u)] = e^{-u\cdot\Xi}\int_{-\infty}^{\infty} f(\xi)\, e^{-\Xi\cdot\xi}d\xi \qquad (3)$$

where C denotes the cumulants, α, β, γ are the order of cumulants, Ξ is the wave number, and $\mathcal{L}$ is the Laplace transformation.

The collision term of the cumulant model is as follows:

$$C^{*}_{\alpha\beta\gamma} = w_{\alpha\beta\gamma} C^{eq}_{\alpha\beta\gamma} + (1 - w_{\alpha\beta\gamma}) C_{\alpha\beta\gamma} \tag{4}$$

where the superscript $*$ means the post-collision state, $w_{\alpha\beta\gamma}$ is the relaxation frequency and $C^{eq}_{\alpha\beta\gamma}$ is the equilibrium cumulant.

(2) Discrete Element Method

The discrete element method is a technique for simulating the behavior of a group of particles by tracking the motion of individual particles. The contact between particles is modeled using springs, dashpots, and friction sliders. In the normal direction, the repulsive force generated by the spring is proportional to the contact depth of the particles, while the damping force generated by the dashpot is proportional to the relative velocity of the particles. In the tangential direction, in addition to these two forces, the friction force due to the relative velocity between particles must also be considered.

However, in this research, the shape of the driftwood is not spherical, so we need to use a method that can handle non-spherical objects in the discrete element method. In this paper, we adopt the approach of representing non-spherical objects as a collection of micro-spherical particles that are rigidly connected [7]. The total contact force acting on the non-spherical objects can be calculated as follows:

$$F_G = \sum^{n} F_i \tag{5}$$

where n is the number of micro-spherical particles.

The translational motion of the non-spherical object can be described by the following equation:

$$m_G \frac{dv_G}{dt} = F_G + m_G g \tag{6}$$

where m_G is the mass of the non-spherical object, v_G is the translational speed of the object and g is the acceleration due to gravity. By integrating Eq. (6) over time, we can calculate the translational motion of the non-spherical object.

The non-spherical object also undergoes rotational motion around its center of mass. Suppose the moment of force is M_G, the position vector of the spherical particle is $\boldsymbol{x}_i$ and the position of barycenter is x_G. Then the moment of force can be calculated as:

$$M_G = \sum^{n} ((\boldsymbol{x}_i - x_G) F_i) \tag{7}$$

$$\frac{dL_G}{dt} = M_G \tag{8}$$

$$w_G = I_G(t)^{-1} L_G \tag{9}$$

where L_G is the angular momentum of the object, w_G is the angular velocity of the object, and $I_G(t)^{-1}$ is the inverse matrix of the moment of inertia tensor of non-spherical particles at time t, it is calculated at each time step because the moment of inertial tensor changes with the rotation of the non-spherical object. $I_G(t)^{-1}$ can be computed by the following equation:

$$I_G(t)^{-1} = R(t) I_G(0)^{-1} R(t)^T \tag{10}$$

where $R(t)$ is the rotation matrix of time t. Using Eq. (8) and Eq. (9) we can get the angular velocity w_G and thus the rotation angle can be updated.

(3) Phase Field Method

The phase field method is a powerful tool for capturing the gas-liquid interface and tracking microstructure evolution in two or three dimensions [8]. In this research, we use a conservative phase field model [9] derived from the Allen-Cahn equation to track the interface.

The Allen-Cahn equation, which is a partial differential equation, is used to model the evolution of a phase field function that describes the interface between two phases. The conservative form of the Allen-Cahn equation is given by:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\vec{u}\phi) = \nabla \cdot \left(M \left(\nabla\phi - \frac{4\phi(1-\phi)}{W} \vec{n} \right) \right) \tag{11}$$

where ϕ is the phase field function, M is the mobility, W is the width of the interface and it is chosen to be $3\Delta x$ in this paper, and $\vec{n}$ is the unit out-normal vector of the interface. The term $\vec{u}$ represents the velocity field, which is used to advect the phase field function.

Suppose the interface located at x_0, in the equilibrium state, the phase field function can be derived from the following equation:

$$\phi_x = \frac{1}{2} \left[1 + \tanh\left(\frac{\psi(x)}{W/2}\right)\right] \tag{12}$$

where $\psi(x) = \boldsymbol{x} - \boldsymbol{x}_0$ is the signed distance function of the gas-liquid interface.

(4) Adaptive Mesh Refinement Method

The adaptive mesh refinement (AMR) method has the ability to allocate high-resolution grids to areas that require high accuracy and low-resolution grids to other areas, thus significantly reducing computation costs compared to uniform meshes.

In this research, we employed a tree-type [10] AMR method using block-structured grids, where the block serves as the smallest entity to assign tasks, and the computational domain is divided into blocks of varying sizes in AMR.

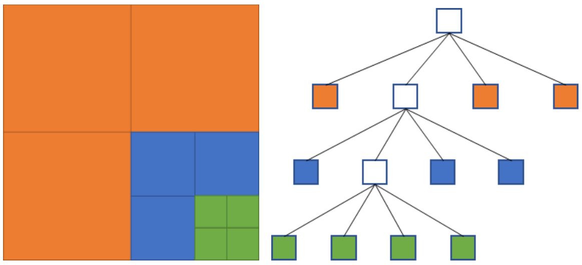

Fig. 1 Tree-type block-structured AMR

Fig. 1 depicts the data structure based on quadtrees (octrees in 3D). A coarse block at a lower level of the quadtree can be subdivided into four (eight in 3D) smaller blocks of equal size at a higher level during the refining process. Similarly, during the coarsening process, four (eight in 3D) child blocks at a higher level can be merged into one parent block at a lower level. It's important to note that the 2:1 balance of neighboring blocks must be maintained, meaning the maximum difference of level between neighboring blocks should not exceed one.

3. RESULTS AND DISCUSSIONS

(1) Simulation when No Driftwood is Released

We collaborated with Dr. Shima from the Sabo & Landslide Technical Center to conduct a numerical simulation in an experimental waterway, as illustrated in Fig. 2. The water flowed from the right side to the left side, passing through a dam before exiting the experiment domain. To investigate the performance of a driftwood trap device, we used the device shown in Fig. 3. The actual length of the device is 37.5m, but it was scaled down to 52.5cm during the experiments, with a distance of 3.5cm between two adjacent trapping devices. The device was placed 7cm before the dam. It is worth noting that both Fig. 2 and Fig. 3 are from Dr. Shima's paper [6].

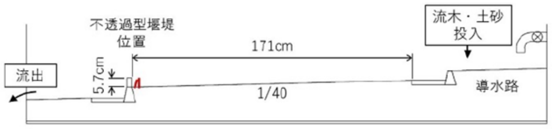

Fig. 2 Experiment water way

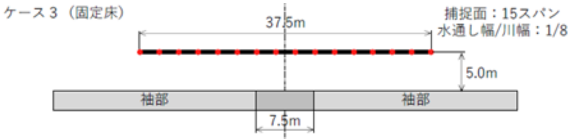

Fig. 3 Driftwood trap device in experiment

Fig. 4 illustrates the simulation conditions of the waterway when no driftwood is released, it is worth noting that the water flows from the left side to the right side in the simulation. The computation domain has a length of 2.04m and a width of 0.85m, with an outflow area length of 0.13m. The inflow velocity is set to 0.25m/s.

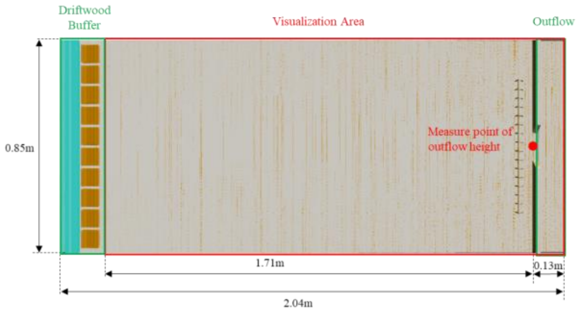

Fig. 4 Simulation condition

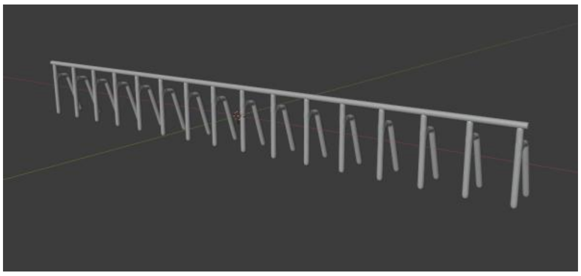

Fig. 5 Driftwood trap device model in simulation

Fig. 5 depicts the 3D model of the driftwood trap device, which was created in Blender with the same details as the physical device shown in Fig. 3. The model was exported as an STL file, converted to a level-set function, and then imported into the simulation code.

The water density used in the simulation is 1000kg/m^3, with a kinematic viscosity of 1.004× $10^6 m^2/s$. The minimum mesh size is 6.64× 10^{-4}m, and the time interval is 7.968× $10^{-5}s$. We use the Dirichlet boundary condition for the inflow and the Neumann boundary condition for the outflow. The simulation is performed using 16 Tesla V100 GPUs on the supercomputer FLOW for a duration of 60 hours.

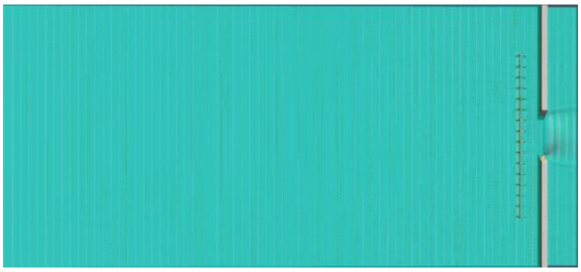

Fig. 6 Simulation snapshot

Fig. 6 shows a simulation snapshot where the water gradually fills the computational domain due to the height difference between the inflow and outflow areas. We also place a measurement point at the outflow area to record the outflow height, and the corresponding profile is shown in Fig. 6.

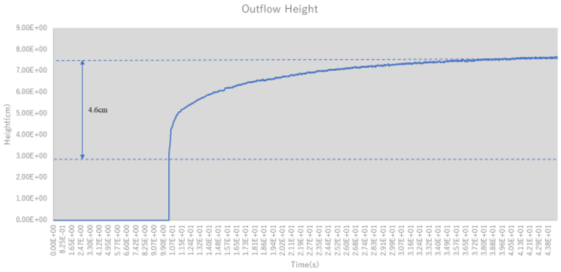

Fig. 7 Outflow height

From Fig. 7, we observe that the outflow height gradually increases initially due to the water flow. As the inflow and outflow reach equilibrium, the outflow height becomes steady. The base height of the outflow area is 4.7cm, as indicated by the lower dashed line in Fig. 6, and the steady height of the outflow area is 8cm, as indicated by the upper dashed line in Fig. 6. This corresponds to a steady water height at the outflow area of 4.6cm, which agrees well with the results reported by Dr. Shima in [6].

(2) Simulation when 100 Pieces of Driftwood are Released

After confirming the inflow and outflow conditions, we released 100 pieces of driftwood into the computational domain. Each driftwood piece was modeled as a cylinder with a radius of 0.2cm and a length of 7cm. The density of driftwood is 850kg/m^3. For the DEM parameters, the Poisson's ratio is 0.33, the restitution coefficient is 0.25 and the friction coefficient is 0.3. Other parameters are the same as the previous simulation.

Figure 8 displays a simulation snapshot capturing the moment when 100 pieces of driftwood reach the trap device. During the simulation, the driftwood escape rate, which indicates the number of pieces of driftwood that pass through the trap device, is found to be 12%. This result is consistent with Dr. Shima's research [6], which reported a driftwood escape rate of 20%, indicating that our simulation aligns well with the experimental findings.

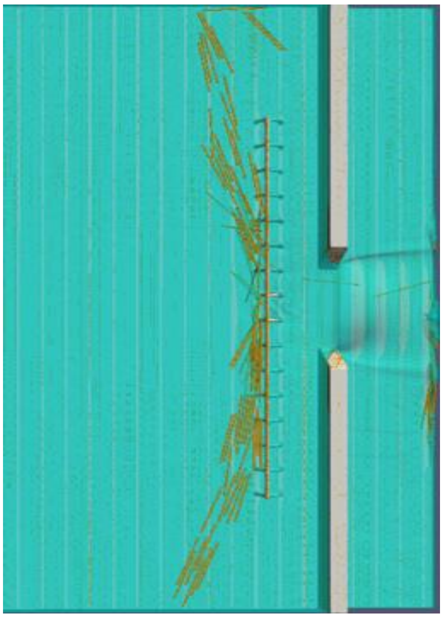

Fig. 8 Enlarged simulation snapshot

The simulation is still ongoing on the FLOW supercomputer, and we are eagerly awaiting additional results to compare with Dr. Shima's findings.

4. CONCLUSIONS

We have developed a high-resolution simulation method for driftwood capture using a cumulant lattice Boltzmann method coupled with adaptive mesh refinement, discrete element, and phase field methods. Our simulation results for both scenarios, with and without driftwood, demonstrate excellent agreement with the actual situations. The significance of this simulation method lies in its ability to aid in the prediction and prevention of driftwood disasters.

ACKNOWLEDGMENT: This research was partly supported by a Grant-in-Aid for Scientific Research (S) 19H05613, from the Japan Society for the Promotion Science (JSPS), and Joint/Research Center for Interdisciplinary Large-scale Information Infrastructures (JHPCN), jh200018 and jh210013, High Performance Computing Infrastructure (HPCI) hp210129 and hp230065 projects, and JST SPRING under grant number JPMJSP2106. The authors thank the Global Scientific Information and Computing Center, Tokyo Institute of Technology for use of the computing resources of the TSUBAME 3.0 supercomputer and the Information Technology Center of Nagoya University for use of the computing resources of the Flow Type II supercomputer.

REFERENCES

[1] Matsutomi H. Method for estimating collision force of driftwood accompanying tsunami inundation flow[J]. Journal of Disaster Research, 2009, 4(6): 435-440.

[2] Sterling J D, Chen S. Stability analysis of lattice Boltzmann methods[J]. Journal of Computational Physics, 1996, 123(1): 196-206.

[3] Geier M, Schönherr M, Pasquali A, et al. The cumulant lattice Boltzmann equation in three dimensions: Theory and validation[J]. Computers & Mathematics with Applications, 2015, 70(4): 507-547.

[4] Berger M J, Oliger J. Adaptive mesh refinement for hyperbolic partial differential equations[J]. Journal of computational Physics, 1984, 53(3): 484-512.

[5] Seiya Watanabe and Takayuki Aoki. Large-scale flow simulations using lattice Boltzmann method with AMR following free-surface on multiple GPUs[J]. Computer Physics Communications, Vol. 264, p. 107871, 2021.

[6] 嶋丈示，安冨懸一. 不透過型砂防堰堤の上流に付設する流木捕捉工の流木捕捉機能[J]. 新砂防= Journal of the Japan Society of Erosion Control Engineering: 砂防学会誌, 2022, 75(4): 14-24.

[7] Markauskas D, Kačianauskas R. Investigation of rice grain flow by multi-sphere particle model with rolling resistance[J]. Granular Matter, 2011, 13(2): 143-148.

[8] Avila-Davila E O, Lopez-Hirata V M, Saucedo-Muñoz M L. Application of Phase-Field Method to the Analysis of Phase Decomposition of Alloys[J]. Modeling and Simulation in Engineering Sciences, 2016: 221.

[9] Chiu P H, Lin Y T. A conservative phase field method for solving incompressible two-phase flows[J]. Journal of Computational Physics, 2011, 230(1): 185-204.

[10] Tiankai T, David R O, Omar G. Scalable parallel octree meshing for terascale applications[C]. Proceedings of the 2005 ACM/IEEE Conference on Supercomputing, SC. 2005, 5: 4.