Coupled Lattice Boltzmann and Discrete Element Simulations of Ship-Ice Interactions

Watanabe, Seiya
Research Institute for Applied Mechanics, Kyushu University

Hu, Changhong
Research Institute for Applied Mechanics, Kyushu University

Aoki, Takayuki
Global Scientific Information and Computing Center, Tokyo Institute of Technology

https://hdl.handle.net/2324/7183292

出版情報：IOP Conference Series: Materials Science and Engineering. 1288, pp.012015-, 2023-08-09. IOP Publishing
バージョン：

IOP Conference Series: Materials Science and Engineering

PAPER • OPEN ACCESS

Coupled Lattice Boltzmann and Discrete Element Simulations of Ship-Ice Interactions

To cite this article: Seiya Watanabe *et al* 2023 *IOP Conf. Ser.: Mater. Sci. Eng.* **1288** 012015

Coupled Lattice Boltzmann and Discrete Element Simulations of Ship-Ice Interactions

Seiya Watanabe[1], **Changhong Hu**[1] **and Takayuki Aoki**[2]

[1] Research Institute for Applied Mechanics, Kyushu University, 6-1 Kasuga-koen Kasuga-city, Fukuoka, Japan

[2] Global Scientific Information and Computing Center, Tokyo Institute of Technology, 2-12-1 i7-3 O-okayama, Meguro-ku, Tokyo, Japan

E-mail: swatanabe@riam.kyushu-u.ac.jp

Abstract. Evaluating ice loads acting on ships is essential for the safety of ships navigating in ice-covered seas. In this study, we develop a CFD method to handle ship, ice, and fluid interaction. The lattice Boltzmann method, capable of large-scale calculations, is applied to the simulation of free-surface fluids. The ice motion is computed by solving the equations of motion of a rigid body, and the discrete element method models the ice-ice and ice-ship contact interactions. A momentum exchange scheme couples the lattice Boltzmann method and particle-based rigid body simulation. We introduce tree-based adaptive mesh refinement and multiple GPU computing to improve grid resolution and computational time. The proposed method is applied to model scale simulations of ship navigation in a brash ice channel. Simulations were performed for various conditions with different ice concentrations and ship velocities, and we observed that ice resistance increased with the ice concentration and the ship velocity increased. The ice motions and resistances obtained from our simulations are reasonable compared to model equations of Finnish-Swedish ice class rules (FSICR) and numerical analyses of a previous study.

1. Introduction

With the shrinking of ice due to global warming, the Arctic Ocean has become more attractive in recent years, and the number of ocean-going commercial vessels using the Northern Sea Route has increased. These ships select routes where it is possible to navigate without ice-breaking capabilities. Since ships collide with ice floes, it is essential to evaluate the ice force acting on the ship, which directly affects the safety of navigation, by experiments and numerical models.

An ice tank experiment using a model ship is an effective measure to predict ice resistance performance. However, facilities with ice tanks are limited to a few places worldwide, and it takes time to fabricate ice channels. Thus ice tank experiments are time-consuming and costly.

Numerical simulation tools that can accurately and quickly predict ice resistance are required to reproduce ice tank experiments. Since interactions between ice, fluid, free surface, and ship are very complex physical phenomena, many studies proposed simulation methods deal only with the ship-ice interaction [1, 2]. Flow fields and free interfaces are not calculated, and floating ice floes are subjected to drag and buoyancy forces modeled by empirical equations. Computational models that handle only ice-ship interactions efficiently estimate ice resistance in broken ice and brash ice channels [3, 4]. However, if the flow field around an ice floe affects other ice floes; for example, if the ice floes move densely together, the accuracy of the ice force estimate is reduced.

Recent performance improvements in computational hardware have enabled computational fluid dynamics (CFD) to calculate even the flow field around ice floes and to treat ice-fluid interactions without empirical modeling. Shafiul Mintu and David Molyneux [5] developed a fluid-solid interaction model combining the smoothed particle hydrodynamics (SPH) and the discrete element method (DEM) for ship-ice interaction in waves. The authors of [6] have developed simulation software that combines the OpenDynamics Engine (ODE) with the lattice Boltzmann method (LBM)-based software ELBM. The GPU, which is high-performance hardware, was used to accelerate the fluid calculations, and their ODE-ELBM software simulates ship-fluid-rigid-ice interactions. The high computational cost of direct ice-fluid interactions still needs to be addressed, limiting the number of floating ice floes in simulations.

In this study, we develop a high-performance CFD tool for ship-fluid-ice interactions using the LBM and the DEM to increase the number of ice pieces handled in simulations. To solve the high computational cost of direct coupling of fluid and ice motions, we introduce the adaptive mesh refinement (AMR) method, which locally assigns a high-resolution grid, and perform parallel computation using multiple GPUs on a supercomputer. The ice floe geometry is represented by a simple model of rigidly-connected micro DEM particles, and all DEM calculations are performed on GPUs. In this paper, we report simulation results of the navigation of a model ship in a brash ice channel composed of small ice cubes. The motion of the ice floes and the ice force acting on the ship are compared with simulations of previous studies and the FSICR formula for validation.

2. Numerical method

We develop a robust and fast computational method for fluid-ship-ice interactions. The single-phase lattice Boltzmann method calculates hydrodynamics with free-surface boundaries. The phase-field method simulates the advection and deformation of the free surface. We apply particle-based rigid body dynamics for the ice simulation and do not consider the fracture and consolidation of the ice floes. A mass-spring-damper system of the DEM computes ice-ice and ice-ship contact interactions. The block-based adaptive mesh refinement method and parallel computing on multiple GPUs speed up simulations.

Our previous studies [7][8] describe the details of the computational method and validation results on benchmark problems. We briefly explain the methodology in this section.

2.1. Lattice Boltzmann method

The lattice Boltzmann method (LBM) assumes that the fluid is virtual particles that collide and stream on a grid. The statistical behavior of the virtual particles follows the Boltzmann equation. By discretizing the Boltzmann equation with the finite velocity direction $\boldsymbol{\xi}_{ijk}$, the lattice spacing Δx, and the time step Δt, the lattice Boltzmann equation is obtained as

$$f_{ijk}\left(\boldsymbol{x}+\boldsymbol{\xi}_{ijk}\Delta t, t+\Delta t\right) = f_{ijk}\left(\boldsymbol{x}, t\right) + \Omega_{ijk}\left(\boldsymbol{x}, t\right). \tag{1}$$

The subscript ijk indicates the discretized velocity direction, which follows the D3Q27 velocity model in this study. A stable and accurate cumulant LBM model [9] is introduced for the collision term Ω to calculate the flow around a ship at high Reynolds numbers.

The fluid density ρ and the flow velocity $\boldsymbol{u}$ are calculated as

$$\rho\left(\boldsymbol{x}, t\right) = \sum_{ijk} f_{ijk}(\boldsymbol{x}, t), \tag{2}$$

$$\rho\boldsymbol{u}\left(\boldsymbol{x}, t\right) = \sum_{ijk} \boldsymbol{\xi}_{ijk} f_{ijk}(\boldsymbol{x}, t). \tag{3}$$

Under the weakly compressible fluid assumption, the fluid pressure is proportional to density as $p = c_s^2\rho$, where c_s is the pseudo sound speed.

2.2. Free-surface model

The free surface shape is captured by a phase-field method that defines order parameter ϕ identifying the liquid and gas phases. We define $\phi = 1$ for the liquid phase and $\phi = 0$ for the gas phase, and the gas-liquid interface varies ϕ smoothly over three lattice widths. The time evolution of the phase-field variable ϕ follows the conservative Allen-Cahn equation,

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\boldsymbol{u}\phi) = \nabla \cdot \left[M \left\{ \nabla \phi - \frac{1 - 4\left(\phi - \phi^{\mathrm{ave}}\right)^2}{W} \boldsymbol{n} \right\} \right], \tag{4}$$

where, W is the interface thickness, and the M parameter sets the interface mobility. ϕ^{ave} is the average value of ϕ between the liquid and gas phases, which in this study is 1/2.

To better handle moving solid boundaries such as a ship and ice floes, we use a phase-field LBM [10] to solve the conservative Allen-Chan equation. For the order parameter ϕ, we define the velocity distribution function h as

$$\phi\left(\boldsymbol{x}, t\right) = \sum_{ijk} h_{ijk}(\boldsymbol{x}, t). \tag{5}$$

h is time updated with a simple collision term using a single relaxation time parameter τ_ϕ, as

$$h_{ijk}\left(\boldsymbol{x} + \boldsymbol{\xi}_{ijk}\Delta t, t + \Delta t\right) = h_{ijk}\left(\boldsymbol{x}, t\right) - \frac{1}{\tau_\phi}\left(h_{ijk}\left(\boldsymbol{x}, t\right) - h_{ijk}^{\mathrm{eq}}\left(\boldsymbol{x}, t\right) \right). \tag{6}$$

h_{ijk}^{eq} is the distribution function of the equilibrium state, given by

$$h_{ijk}^{\mathrm{eq}} = \phi \Gamma_{ijk} + \theta w_{ijk} \boldsymbol{\xi}_{ijk} \cdot \boldsymbol{n}, \tag{7}$$

$$\Gamma_{ijk} = w_{ijk} \left\{ 1 + \frac{\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u}}{c_{\mathrm{s}}^2} + \frac{\left(\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u}\right)^2 - c_{\mathrm{s}}^2 \left|\boldsymbol{u}\right|^2}{2c_{\mathrm{s}}^4} \right\}, \tag{8}$$

$$\theta = \frac{M}{c_{\mathrm{s}}^2} \left\{ \frac{1 - 4\left(\phi - \phi^{\mathrm{ave}}\right)^2}{W} \right\}, \tag{9}$$

where, w_{ijk} is the weight parameter of the D3Q27 model. The relaxation parameter τ_ϕ is determined from the interface mobility M as

$$\tau_\phi = \frac{1}{2} + \frac{M}{c_s^2 \Delta t}. \tag{10}$$

The cumulant LBM calculates only the water domain ($\phi \geq \phi^{\mathrm{ave}}$) and imposes boundary conditions on the free surface for computational stability. The velocity distribution function f for the gas ($\phi < \phi^{\mathrm{ave}}$)is undefined. At grid points near the free surfaces, missing distribution functions from the gas phase are given as

$$f_{ijk}\left(\boldsymbol{x}, t + \Delta t\right) = f_{ijk}^{\mathrm{eq}}\left(\rho_0, \boldsymbol{u}\right) - \left[f_{\overline{ijk}}\left(\boldsymbol{x}, t\right) - f_{\overline{ijk}}^{\mathrm{eq}}\left(\rho_0, \boldsymbol{u}\right) \right], \tag{11}$$

$$f_{ijk}^{\mathrm{eq}}\left(\rho_0, \boldsymbol{u}\right) = w_{ijk}\rho_0 \left[1 + 3\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u} + \frac{9}{2}\left(\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u}\right)^2 - \frac{3}{2}\boldsymbol{u} \cdot \boldsymbol{u} \right], \tag{12}$$

where, $f_{\overline{ijk}}$ is the distribution function in the opposite direction of f_{ijk}, and ρ_0 is the reference density.

2.3. Ice model

A particle-based rigid body simulation method [11] calculates ice motions and ice-ice, ice-ship contact interactions. A rigid body consisting of many small spherical particles represents the shape of a ship or floating ice floes. The particles, placed on a solid surface, detect collisions with other objects and calculate the contact force based on the spring and dashpot models of the DEM [12]. The DEM has been applied in many numerical studies of ice-structure interactions[13]. The spring applies a repulsive force proportional to the penetration depth of the particles, and the dashpot adds a damping force that depends on the relative velocity of the particles. The friction between the particles is expressed by setting an upper bound on the tangential force according to Coulomb's law of friction. Ice accretions are not considered. A sum of contact forces $\boldsymbol{F}_i^{\mathrm{DEM}}$ acting on i-th spherical particle gives the total force acting on the rigid body and the torque around the center of gravity.

The motion of a rigid body is calculated from two parts: translation and rotation. The equation of motion for the translation of a rigid body is

$$m_s \frac{\mathrm{d}\boldsymbol{v}_s}{\mathrm{d}t} = \sum_i \boldsymbol{F}_i^{\mathrm{DEM}} + \boldsymbol{F}_s^{\mathrm{fluid}} + m_s \boldsymbol{g}, \tag{13}$$

where m_s and $\boldsymbol{v}_s$ are the rigid body's mass and translation velocity, respectively. The equation of motion for rigid body rotation is

$$\frac{\mathrm{d}\boldsymbol{L}_s}{\mathrm{d}t} = \sum_i \boldsymbol{r}_i \times \boldsymbol{F}_i^{\mathrm{DEM}} + \boldsymbol{M}_s^{\mathrm{fluid}}, \tag{14}$$

$$\boldsymbol{\omega}_s(t) = \boldsymbol{I}_s(t)^{-1} \boldsymbol{L}_s(t), \tag{15}$$

where, $\boldsymbol{L}_s$ is the angular momentum of the rigid body, $\boldsymbol{\omega}_s$ is the angular velocity, and $\boldsymbol{I}_s$ is the inertia tensor. $\boldsymbol{r}_i$ is the relative position of the i-th particle to the center of gravity of the rigid body. $\boldsymbol{F}_s^{\mathrm{fluid}}$ and $\boldsymbol{M}_s^{\mathrm{fluid}}$ are the force and torque given by the LBM fluid and are explained in the next section. These equations of motion are integrated over time using the first-order Euler method.

2.4. Fluid-structure interaction

The mesh for the LBM calculations is fine enough for the size of the ice floes, and the fluid-ice interactions are calculated directly from the flow field around the ice. We use a coupling method based on momentum exchange between objects and fluid. The motion of the ice pieces is treated as a moving boundary condition in the LBM. The ice shape is implicitly represented by a signed distance function, the distance from the ice surface to the grid point. A non-slip moving boundary condition based on the interpolated bounce-back scheme [14] is used at the ice-fluid interface. According to the normalized signed distance function q, the velocity distribution function bouncing off the object surface is calculated as

$$f_{ijk}(\boldsymbol{x}, t+\Delta t) = \begin{cases} 2q f_{\overline{ijk}}(\boldsymbol{x}, t) + (1-2q) f_{\overline{ijk}}(\boldsymbol{x} + \boldsymbol{\xi}_{ijk}\Delta t, t) + \\ \qquad \dfrac{6\omega_{ijk}\rho\left(\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u}_{\mathrm{wall}}\right)}{c^2} & \text{if } 0 < q \leq 0.5 \\ \dfrac{1}{2q}\left\{ f_{\overline{ijk}}(\boldsymbol{x}, t) + \dfrac{6\omega_{ijk}\rho\left(\boldsymbol{\xi}_{ijk} \cdot \boldsymbol{u}_{\mathrm{wall}}\right)}{c^2} \right\} + \\ \qquad \dfrac{(2q-1)}{2q} f_{ijk}(\boldsymbol{x}, t) & \text{if } 0.5 < q \leq 1.0, \end{cases} \tag{16}$$

where, $\boldsymbol{u}_{\mathrm{wall}}$ is the wall velocity.

The force exerted on an object by a distribution function bouncing off an object's surface is calculated as

$$\boldsymbol{F}_{ijk}\left(\boldsymbol{x},t\right)=\left(\boldsymbol{\xi}_{ijk}-\boldsymbol{u}_{\mathrm{wall}}\right)f_{ijk}\left(\boldsymbol{x},t\right)-\left(\boldsymbol{\xi}_{\overline{ijk}}-\boldsymbol{u}_{\mathrm{wall}}\right)f_{\overline{ijk}}\left(\boldsymbol{x},t+\Delta t\right). \tag{17}$$

This method calculates the momentum exchange considering the object velocity to satisfy Galilean invariance [15]. By summing $\boldsymbol{F}_{ijk}$ at the object surface region Ω_s, the total fluid force $\boldsymbol{F}_s$ and torque $\boldsymbol{M}_s$ is obtained as

$$\boldsymbol{F}_s\left(t\right)=\sum_{\boldsymbol{x}\in\Omega_s}\sum_{ijk\in q^*}\boldsymbol{F}_{ijk}\left(\boldsymbol{x},t\right), \tag{18}$$

$$\boldsymbol{M}_s\left(t\right)=\sum_{\boldsymbol{x}\in\Omega_s}\left\{\left(\boldsymbol{x}-\boldsymbol{x}_s\right)\times\sum_{ijk\in q^*}\boldsymbol{F}_{ijk}\left(\boldsymbol{x},t\right)\right\}, \tag{19}$$

where, q^* denotes the set of distribution functions that bounced off the surface. To stabilize the computation of fluid-solid couplings, $\boldsymbol{F}_s^{\mathrm{fluid}}$ and $\boldsymbol{M}_s^{\mathrm{fluid}}$, which are substituted into the equations of motion (13)(14), are the average values of $\boldsymbol{F}_s$ and $\boldsymbol{M}_s$ of the previous and current time steps [16] as

$$\boldsymbol{F}_s^{\mathrm{fluid}}=\frac{1}{2}\left\{\boldsymbol{F}_s\left(t\right)+\boldsymbol{F}_s\left(t-\Delta t\right)\right\}, \tag{20}$$

$$\boldsymbol{M}_s^{\mathrm{fluid}}=\frac{1}{2}\left\{\boldsymbol{M}_s\left(t\right)+\boldsymbol{M}_s\left(t-\Delta t\right)\right\}. \tag{21}$$

Moving boundary conditions also apply to the ship-fluid boundaries as well as to the ice. In this study, the ship velocity is constant, so the effect is one-way from the ship to the fluid.

2.5. Adaptive mesh refinement (AMR)

To reduce the computational cost (corresponding to the number of computational grids), we introduce an adaptive mesh refinement method (AMR method) in the LBM and the phase-filed computation. The AMR method allocates a high-resolution grid in a local part of the computational domain. Figure 1 shows that the octree data structure recursively divides the computational domain into cubic regions called blocks. Our AMR strategy places fine blocks near ice pieces, ships, water surfaces, and eddies. Coarse blocks are away from the refinement area. A tree structure corresponds to a cubic region; therefore, the forest of octree data structure, which arranges multiple octree structures, subdivides non-cubic computational domains.

In our simulation code, almost all computational subroutines, such as the LBM, the interface capture, and the DEM, are executed on the GPU; the CPU handles mesh generation and data output. To efficiently parallelize threads on the GPU, a block-structured mesh allocates 8 × 8 × 8 computational cells to each block divided by an octree structure. The calculation points of the LBM are placed at the centers of the computational cells, so-called cell-centered meshes.

At the boundaries where the resolution changes, the velocity distribution function is coarsened and refined using trilinear interpolation. The non-equilibrium components are not rescaled.

Multi-GPU computing is necessary for faster and larger-scale computations. For efficient parallel computing, load balancing is essential to equalize the computational cost of each GPU as much as possible. In this study, we use the Morton curve, one of the space-filling curves, to address the load balancing problem. A Morton curve divides the computational domain to equal the number of computational blocks. A single MPI process and its corresponding GPU handle the computation in one of the divided subdomains.

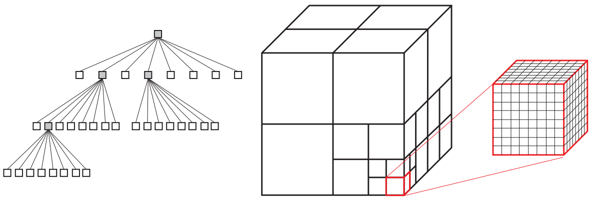


Figure 1. Block-structured mesh based on octree data structure. Each block has 8 × 8 × 8 cells.

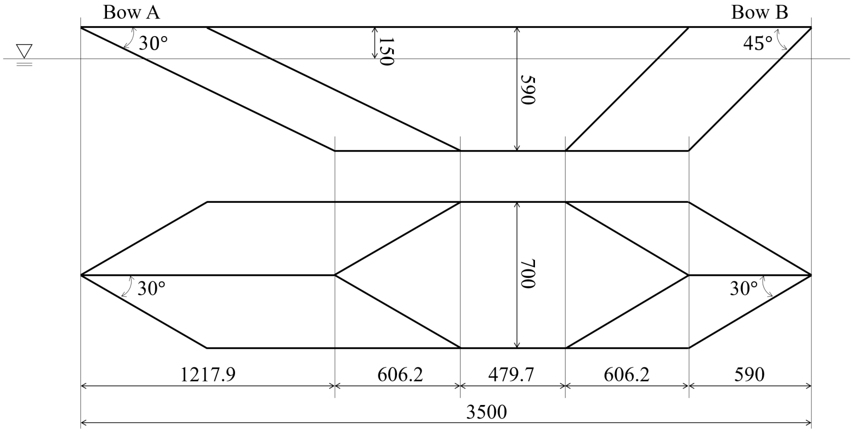


Figure 2. Geometry of model ship (Upper: side view, Lower: bottom view).

Calculation points near the subdomain boundaries access data on the GPU handling the neighboring subdomain. Therefore, data communication with the neighboring GPUs occurs every time step. Some computer systems allow direct data transfer between GPUs via CUDA-aware MPI or NVLink. To ensure that the simulation code does not depend on a specific GPU system, however, in our code, the data transfer between GPUs is realized by the CPU.

3. Simulation setup

We evaluate the ice resistance of ship navigation in a brash ice channel by model-scale simulations. The computational conditions were set up regarding numerical tests conducted by Konno et al. [4]. The model ship B-063 of the National Maritime Research Institute was used. This model ship has a simple shape for ice-breaking experiments. As shown in Figure 2, it is a hexagonal pyramidal trapezoidal shape with a total length of 3.5 m and a width of 0.7 m. The bow at both ends of the hull can be used for experiments. One side has a bow inclination of 30 degrees (Bow A), and the other has a bow inclination of 45 degrees (Bow B).

The ice class was 1A according to the Finnish-Swedish Ice Class Rules (FSICR) [17]. The scale ratio of the model to the full-scale ship was set to 25. The average thickness of the brash ice channel in the model scale was 41 mm (corresponding to 1 m at full-scale) based on the FSICR guidelines.

The computational domain was 11.15 m × 1.39 m × 1.39 m, and the water depth was set at 1.11 m. A brash ice channel was modeled by floating 41 mm ice cubes at equal intervals. The width and length of the brash ice channel were 1.39 m and 5.56 m, respectively. Figure 3 shows the arrangement of the model ship and the brash ice channel in the computational domain. The ice density was 950 kg/m^3. We investigated the effect of ice concentration on ice resistance in three cases with ice concentrations of 50%, 68%, and 88%, corresponding to 2034, 3136, and 4096 ice cubes, respectively.

The full-scale ship velocity in a brash ice channel was 5 knots according to FSICR, and the model-scale ship velocity was determined by the similarity rule to be 1 knot = 0.514 m/s. For the two brash ice channels with ice concentrations of 50% and 88%, the ship velocity was set

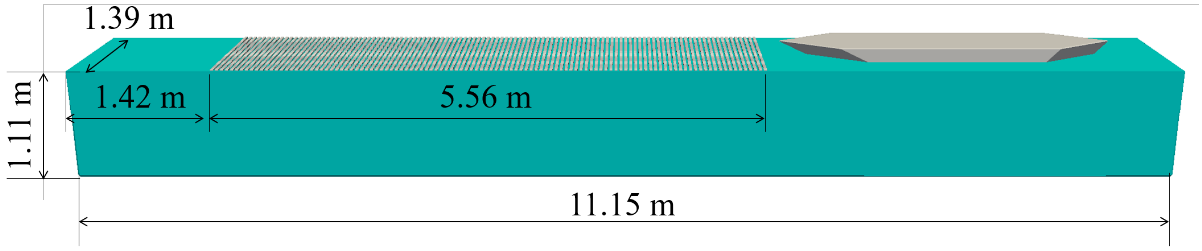
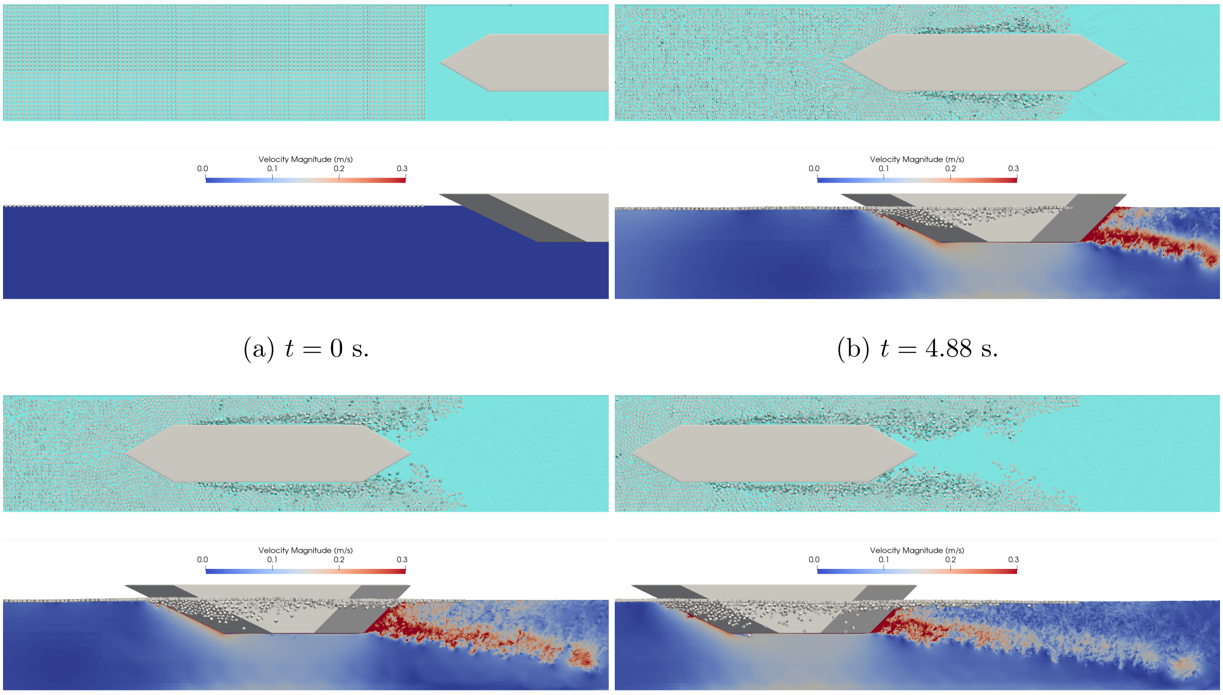


Figure 3. Initial arrangement of model ship and brash ice channel in computational domain.



(a) $t = 0$ s.

(b) $t = 4.88$ s.

(c) $t = 7.32$ s.

(d) $t = 9.75$ s.

Figure 4. Snapshots of ship-ice interaction simulation with navigation speed 0.514 m/s and ice concentration 68%. Bottom panels show velocity profile.

to 1 knot. Three cases with different ship velocities of 1, 1.5 and 2 knots in model scale were run for the 68% concentration to study the effect of ship velocity on the ice resistance and the motion of ice cubes.

The finest mesh had a resolution of 2.72 mm and was placed near the free surface, the ship, the ice cubes, and the eddies. Approximately 15 meshes were per side of 41 mm of the ice cube. The coarsest grid was 43.4 mm. At initialization, the number of grid points was about 160 million, about 15% of that using a uniform mesh. The calculations were performed on 32 GPUs of NVIDIA Tesla P100 and were completed within 24 hours in each case.

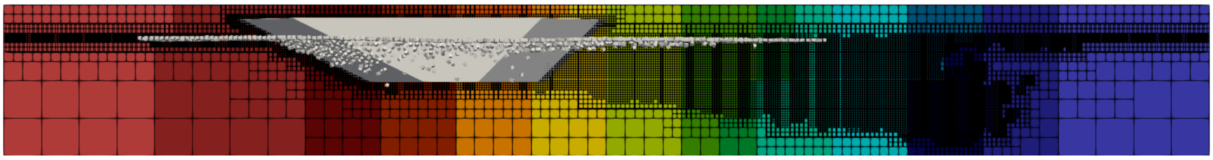
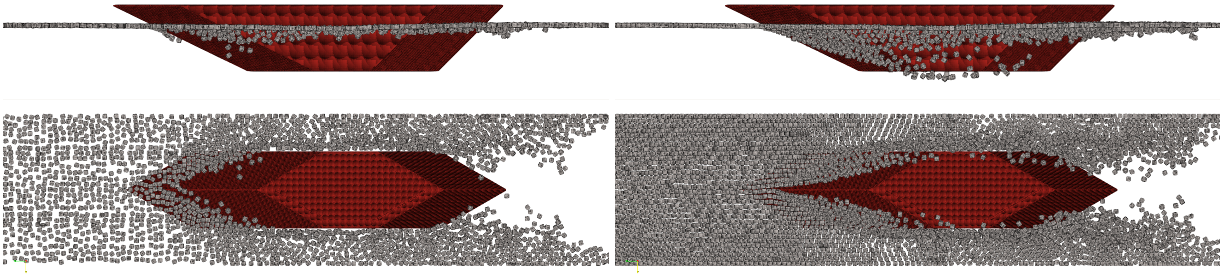


Figure 5. Computational blocks for ship-ice interaction simulation. The color indicates domain decomposition for multi-GPU computation.



(a) 50%.

(b) 88%.

Figure 6. Comparison of ice motions with different ice concentrations of channel.

4. Results and discussion

4.1. Ice motion

As an example of the simulation results, Figure 4 shows simulation snapshots under the ship velocity of 0.514 m/s and the ice concentration of 68%. The top panels are a top view visualizing the free surface, ice, and ship, and the bottom panels are a side view showing the ice, ship, and velocity profile in the center of the channel. Ice cubes colliding with the ship were either pushed to the side of the hull or sank to the bottom. The sinking ice cubes floated to the side of the ship. An ice-free channel was formed in the ship trace. Simulations of previous studies [4] have confirmed similar ice behavior. In addition, the ship's navigation generated waves that induced the heaving motion of the ice pieces. Figure 5 shows the computational block. Due to the eddy, the blocks were divided into smaller blocks in the ship wake. The colors indicate the domain partitioning for multiple GPU computations.

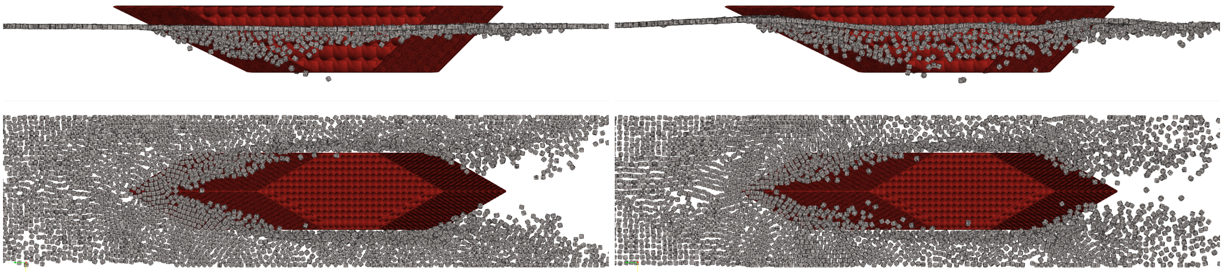
Figure 6 shows the simulated ice motions for different ice concentrations in the brash ice channel. As the ice concentration increased, more ice pieces sank deeper into the lower part of the hull. The width of the ice-free channel narrowed as the ice concentration increased.

Figure 7 shows ice motion differences due to ship velocity. The ship velocity did not significantly affect the sinking motion of the ice cubes. The width of the ice-free channel formed behind the ship narrows with increasing velocity. Faster ship velocity generated higher ship waves; thus, the heaving motion of ice pieces tended to increase with the increase in ship velocity.

Finally, we check the effect of bow shape on ice motion. Figure 8 shows the simulation results for two bow shapes, Bow A and Bow B, at a ship speed of 0.514 m/s and 68% concentration. Bow A, which has a shallower bow angle, has more ice pieces under the hull than Bow B. Bow B has a wider ice-free channel behind the ship.

4.2. Ice resistance

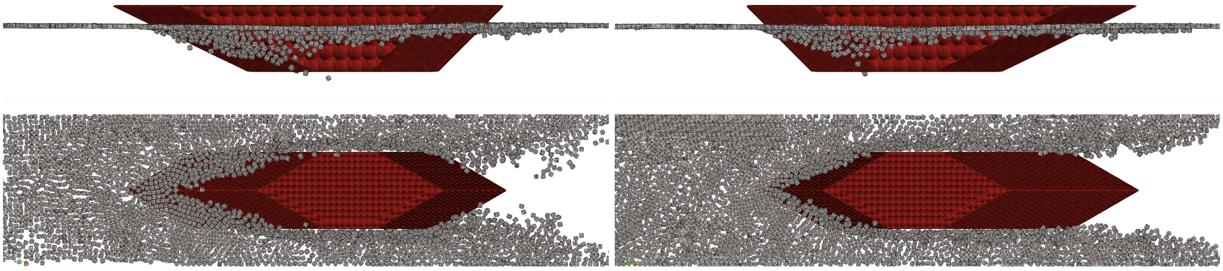
We evaluate ice resistance in a brash ice channel at model ship scale. Ice resistance was calculated by summing the contact forces between the ice cubes and the ship calculated by the DEM model



(a) 0.512 m/s (1 knots).

(b) 1.024 m/s (2 knots).

Figure 7. Comparison of ice motions with different ship velocities.



(a) Bow A.

(b) Bow B.

Figure 8. Comparison of ice motions with different bow angles.

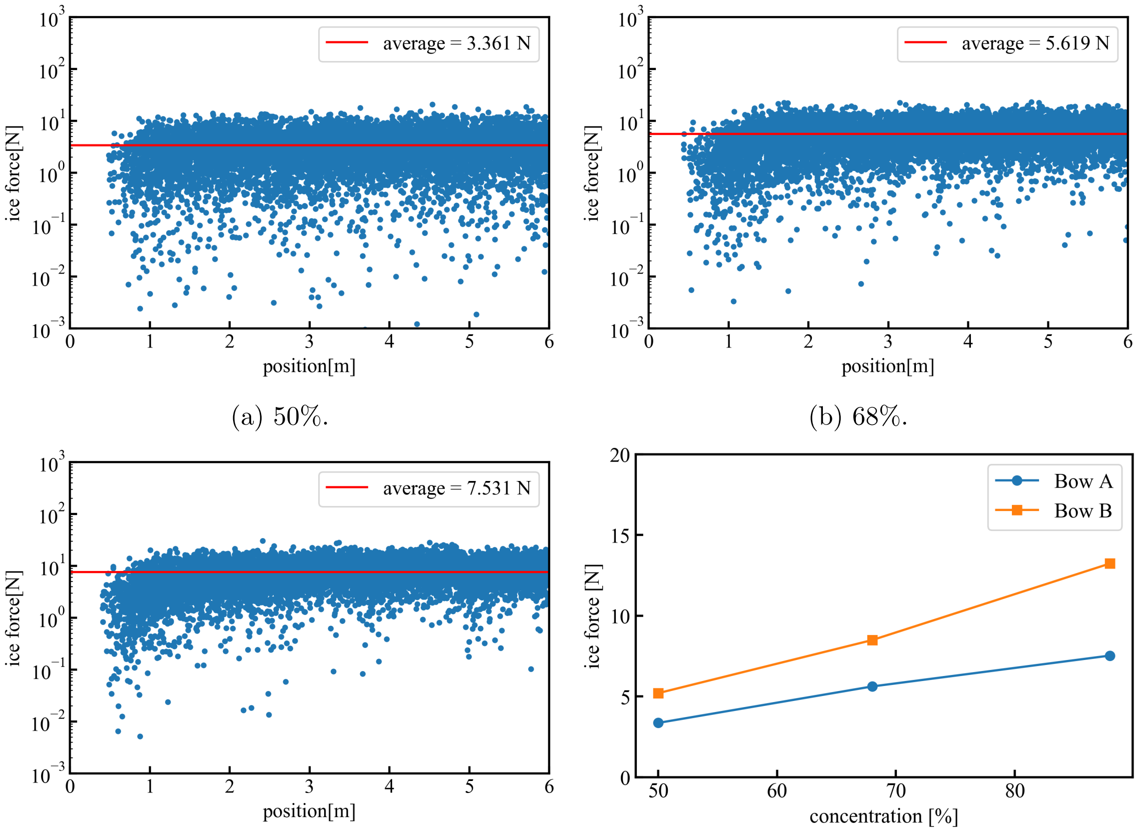
at the ship's surface. Ice resistance did not contain water resistance.

The time histories of ice force for different ice concentration levels are shown in Figure 9. The ship velocity was 0.512 m/s. Figure 9 (a), (b), and (c) show the ice forces on a log scale, with the horizontal axis representing the ship position. The ice forces varied widely on the order of 10^{-3} N to 10^{2} N. In previous studies [6], such oscillations in ice force have been reported in CFD simulations. The ice load oscillations were smaller for higher ice concentrations. The number of ice pieces in contact with the ship affects the load oscillation. The higher the concentration, the smaller the variation in the number of pieces in contact and, therefore, the smaller oscillations in ice force. The red lines in Figure 9 (a), (b), and (c) show the time-averaged ice force, and Figure 9 (d) shows the relationship between ice concentration and time-averaged ice resistance. It was observed that the ice force increased with increasing ice concentration.

The time series of ice forces in a channel with 68% ice concentration with different ship velocities are shown in Figure 10. The ice force tends to increase as the ship velocity increases. A gradual load change can be observed at the velocity of 2 knots. We consider that this is due to the heave motions of the ice pieces caused by the ship waves and the change in the ice contacting area on the ship.

Figure 9 (d) and Figure 10 (d) compare the ice resistance using Bow A and Bow B. The ice resistance was larger when using Bow B, which has a greater bow angle, than Bow A.

The real-scale ice resistances were estimated based on the cubic law (scale ratio was 25) and



(a) 50%.

(b) 68%.

(c) 88%.

(d) Mean ice force.

Figure 9. Relationship between ice force and concentration. Force signals of bow A are shown.

compared with the ice resistance R_{CH} estimated by the FSICR formula [17]

$$R_{\mathrm{CH}} = C_1 + C_2 + C_3 C_\mu \left(H_{\mathrm{F}} + H_{\mathrm{M}}\right)^2 \left(B + C_\psi H_{\mathrm{F}}\right) + C_4 L_{\mathrm{PAR}} H_{\mathrm{F}}^2 + C_5 \left(\frac{LT}{B^2}\right)^3 \frac{A_{\mathrm{wf}}}{L}, \tag{22}$$

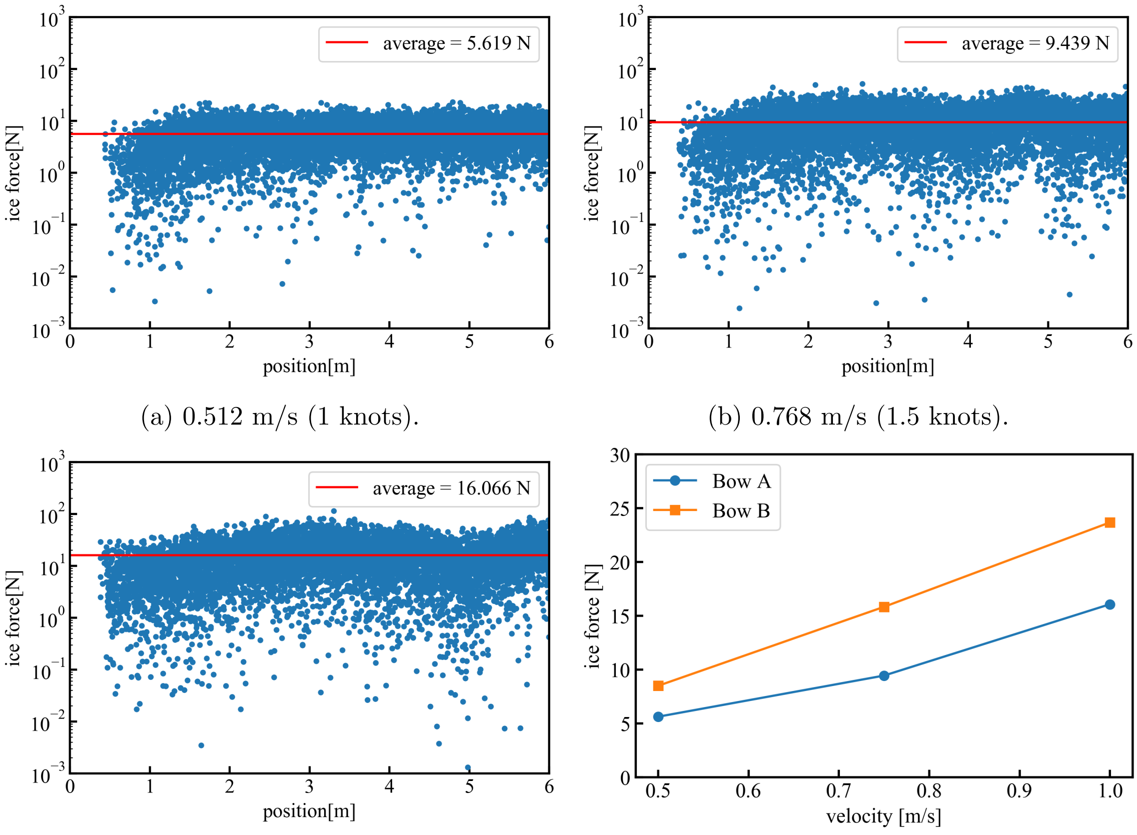
$$C_\mu = 0.15 \cos\phi_2 + \sin\psi \sin\alpha, \tag{23}$$

$$C_\psi = 0.047 \cdot \psi - 2.115, \tag{24}$$

$$H_{\mathrm{F}} = 0.26 + \left(H_{\mathrm{M}} B\right)^{0.5}, \tag{25}$$

$$\psi = \arctan\left(\frac{\tan\phi_2}{\sin\alpha}\right). \tag{26}$$

For ice class 1A, the resistance C_1 and C_2 for breaking the consolidated layer were zero. The resistance coefficients due to friction between the brash ice and the ship were set to $C_3 = 845$ kg/(m^{2}s^2), $C_4 = 42$ kg/(m^{2}s^2), and $C_5 = 825$ kg/s^2. Table 1 shows the dimensions of the ship and some other parameters for the FSICR formula. Table 2 compares the simulation results and the FSICR formula of ice resistance in a class 1A brash ice channel. Previous studies have reported that the FSICR formula tends to estimate ice resistance larger than that of the ice tank experiment [18]. Our simulation results were reasonable because the order of the ice forces was consistent with the FSICR equation, and the simulated ice forces were smaller than the FSICR formula. However, when comparing the ratio of resistance between Bow A



(a) 0.512 m/s (1 knots).

(b) 0.768 m/s (1.5 knots).

(c) 1.024 m/s (2 knots).

(d) Mean ice force.

Figure 10. Relationship between ice force and ship velocity. Force signals of bow A are shown.

Table 1. Dimensions of model ship and some other parameters for FSICR formula.

Length of the ship between the perpendiculars [m]	L	76.0
Length of the parallel midship body [m]	L_{PAR}	45.7
Maximum breadth of the ship [m]	B	17.5
Actual ice class draughts of the ship [m]	T	11.0
Area of the waterline of the bow [m^2]	A_{wf}	96.25
Angle of the waterline at B/4 [degrees]	α	30
Rake of the stem at the centreline [degrees]	ϕ_1	30 for Bow A, 45 for Bow B
Rake of the bow at B/4 [degrees]	ϕ_2	30 for Bow A, 45 for Bow B
Thickness of the brash ice in mid channel [m]	H_{M}	1

and Bow B cases, the simulation result was 1.76, which was different from 1.21 in the FSICR formula. Although more detailed and accurate validation tests, such as comparisons with tank experiments, are needed, even from the current results, we consider that the proposed ship-ice interaction simulation method helps evaluate ice force in brash ice channels.

Table 2. Comparison of ice resistance between simulation and FSICR equation on real scale.

	Present	FSICR
Bow A	188 kN	293 kN
Bow B	330 kN	355 kN

5. Conclusion

We have developed a numerical method for ship-ice interactions using the LBM and the DEM. The proposed method was applied to model-scale simulations for ship navigation in a brash ice channel to validate the ice piece motions and ice resistance. The simulation reproduced appropriate ice motions, such as the ice pieces sinking under the hull and forming an ice-free channel behind the ship. The ice forces acting on the ship increased with increasing channel ice concentration and ship speed. The simulated ice loads were in the same order of magnitude as the ice forces estimated by the FSICR equation. We conclude that the proposed model was adequate for evaluating ice motions and forces based on the validation results.

Future work includes more detailed and accurate validation compared to ice tank experiments.

Acknowledgements

This research was partly supported by KAKENHI, Grant-in-Aid for Scientific Research (S) JP19H05613, JP22K14181 from Japan Society for the Promotion of Science (JSPS) and 'Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures (JHPCN)' jh210031. The numerical calculations were carried out using the computing resources of the TSUBAME3.0 supercomputer at Tokyo Institute of Technology.

References

[1] Xue Y, Liu R, Li Z and Han D 2020 *Ocean Engineering* **215** 107853
[2] Li F and Huang L 2022 *Journal of Marine Science and Engineering* **10**(2) 165
[3] hwan Kim J, Kim Y, soo Kim H and yeob Jeong S 2019 *Ocean Engineering* **180** 162–174
[4] Akihisa K and Yoshimoto K 2009 *Journal of the Japan Society of Naval Architects and Ocean Engineers* **10** 49–56
[5] Mintu S and Molyneux D 2018 *OTC Arctic Technology Conference* (OnePetro)
[6] Janßen C F, Mierke D and Rung T 2017 **155** 22–32
[7] Watanabe S and Aoki T 2021 *Computer Physics Communications* **264** 107871
[8] Watanabe S, Fujisaki S and Hu C 2021 *Journal of Hydrodynamics* **33**(2) 185–194
[9] Geier M, Schönherr M, Pasquali A and Krafczyk M 2015 *Computers & Mathematics with Applications* **70**(4) 507–547
[10] Geier M, Fakhari A and Lee T 2015 *Physical Review E* **91**(6) 063309
[11] Bell N, Yu Y and Mucha P J 2005 *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* SCA '05 (New York, NY, USA: Association for Computing Machinery) p 77–86
[12] Cundall P A and Strack O D 1979 *Geotechnique* **29**(1) 47–65
[13] Tuhkuri J and Polojärvi A 2018 *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **376**(2129) 20170335
[14] Bouzidi M, Firdaouss M and Lallemand P 2001 *Physics of fluids* **13**(11) 3452–3459
[15] Wen B, Zhang C, Tu Y, Wang C and Fang H 2014 *Journal of Computational Physics* **266** 161–170
[16] Aidun C K, Lu Y and Ding E J 1998 *Journal of Fluid Mechanics* **373** 287–311
[17] THE STRUCTURAL DESIGN AND ENGINE OUTPUT REQUIRED OF SHIPS FOR NAVIGATION IN ICE "FINNISH-SWEDISH ICE CLASS RULES" `https://www.sjofartsverket.se/globalassets/isbrytning/pdf-regelverk/finnish-swedish_iceclass_rules.pdf` last accessed 18 April 2023
[18] Hellmann J H, Rupp K H and Kuehnlein W L 2005 *24th International Conference on Offshore Mechanics and Arctic Engineering: Volume 2* pp 923–930