Proceedings of the Conference on Computational Engineering and Science JSCES
Vol. 30, Jun. 2025

GPU-Accelerated Multiphase Flow Solver Employing an Improved Particle Level Set Method with Kernel Function Correction for Complex Interface Dynamics

Shunsuke Kurioka [1] and Changhong Hu [2]

1) *Ph.D student, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University (6-1 Kasuga-koen, Kasuga 816-8580, Japan, E-mail: 3ES24030G@s.kyushu-u.ac.jp)*

2) *Ph.D, Professor, Research Institute for Applied Mechanics, Kyushu University (6-1 Kasuga-koen, Kasuga 816-8580, Japan, E-mail: hu@riam.kyushu-u.ac.jp)*

This study aims to enhance the accuracy of multiphase flow simulations and enable large-scale computations by integrating the Improved Particle Level Set method with high-order Kernel function Correction (PLS/KC) into a fluid solver and developing a GPU-accelerated parallel computing framework. The PLS/KC method improves interface accuracy and mass conservation by placing particles near the interface and applying corrections using high-order kernel functions. We implement the PLS/KC method into a GPU-compatible fluid solver using FORTRAN90 and OpenACC and evaluate its performance in large-scale parallel computations. Benchmark tests demonstrate that a single GPU achieves up to 11.1 times speedup compared to a 120-core CPU, significantly enhancing computational efficiency. Additionally, the solver supports large-scale simulations with up to 150 million grid points, confirming its effectiveness for practical multiphase flow analysis.

Key Words: *GPU computing, Multiphase flow, High-order interface capturing, Particle level set*

1. INTRODUCTION

The numerical simulation of multiphase flows is a crucial issue in various fields, including ocean engineering and fluid machinery. In particular, for multiphase flow analyses involving large interfacial deformations, it is essential to accurately capture complex interface shape changes. To address this challenge, various interface capturing methods, such as the Volume of Fluid (VOF) method and the Level Set (LS) method, have been proposed. The LS method, introduced by Osher and Sethian (1988) [1], represents the interface implicitly using a signed distance function, facilitating the handling of topological changes and curvature calculations. The Particle Level Set (PLS) method, developed by Enright et al. (2002) [2], combines the LS method with Lagrangian particles to improve mass conservation and shape preservation. Although the PLS method is computationally efficient and easily applicable to three-dimensional problems, its performance depends on the number and distribution of particles, necessitating careful design.

To address these issues, the authors have proposed an improved PLS method, referred to as the PLS/KC method, which enhances accuracy and mass conservation by correcting particle position information using a high-order kernel function [3]. This method is characterized by the placement of particles on the interface and the refinement of interfacial particle positions through a high-order kernel function to improve interface accuracy.

In this study, a multiphase flow solver incorporating the PLS/KC method as the interface capturing scheme is developed as an in-house code. The fluid solver is implemented using FORTRAN90 with OpenACC to enable GPU computing, thereby facilitating large-scale and high-speed simulations. The computational performance of the GPU-based multiphase flow solver is validated using a practical dam-break flow problem.

2. GOVERNING EQUATIONS FOR FLUID SIMULATION

In this study, a multiphase flow simulation code was developed by integrating a fluid dynamics solver based on the incompressible Navier–Stokes equations with an interface capturing module employing the Level Set method. This computational framework adopts a Cartesian coordinate system and employs numerical discretization using the finite difference method on a uniformly spaced orthogonal grid.

The incompressible Navier–Stokes equations are expressed as follows:

$$\nabla \cdot \boldsymbol{u} = 0 \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\frac{\nabla p}{\rho} + \frac{1}{\rho}\nabla \cdot \left(\mu(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T)\right) + \boldsymbol{f} \tag{2}$$

Here, $\boldsymbol{u}$ represents the fluid velocity, t denotes time, p is the pressure, ρ is the density, μ is the viscosity coefficient, and $\boldsymbol{f}$ represents the body force per unit mass. The fractional step method is employed to separate the advection and non-advection terms in the numerical solution. The Navier–Stokes

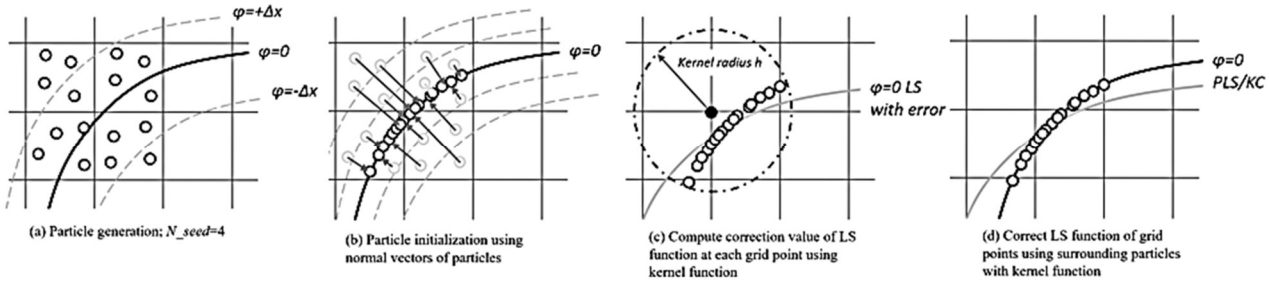

Fig. 1. Schematic diagram of the PLS/KC method [3]

equations are discretized using the finite difference method. The advection term is discretized using a third-order Weighted Essentially Non-Oscillatory (WENO) scheme, while the viscous term (second derivative term) is discretized using a second-order central difference scheme. The pressure Poisson equation is solved using the Preconditioned BiCGStab (PBiCGStab) method, with a convergence criterion set to 10^{-10}. The pressure term is treated implicitly, whereas all other terms are treated explicitly. For time integration, a third-order Total Variation Diminishing (TVD) Runge-Kutta scheme is employed.

The density ρ and viscosity coefficient μ within the computational domain are defined as follows:

$$\rho = \left(1 - H_{vof}\right) \rho_{fluid} + H_{vof}\, \rho_{fluid} \tag{3}$$

$$\mu = \left(1 - H_{vof}\right) \mu_{flu} + H_{vof}\, \mu_{fluid2} \tag{4}$$

Here, H_{vof} is a Heaviside function that takes values in the range of 0 to 1 and behaves similarly to the Volume of Fluid (VOF) function. The Heaviside function is defined as follows [13]:

$$H_{vof} = \begin{cases} 0 & if\ \varphi < -\beta \\ \frac{1}{2}\left[1 + \frac{\varphi}{\beta} + \frac{1}{\pi}\sin\left(\frac{\pi}{\beta}\varphi\right)\right] & if\ \varphi \le |\beta| \\ 1 & if\ \varphi > +\beta \end{cases} \tag{5}$$

The Level Set function ϕ is transformed into a function analogous to the VOF function and is coupled with the Navier–Stokes equations at each time step during time integration.

3. INTERFACE TRACKING FOR MULTIPHASE FLUID

As the interface capturing scheme for multiphase flows, this study employs an improved Particle Level Set method (PLS/KC) [3], which is based on the conventional PLS method [2][4][5] and enhances both interface reconstruction accuracy and mass conservation. The PLS/KC method is characterized by (1) an improved Lagrangian particle placement strategy and (2) an advanced correction method for the Level Set function. In the PLS/KC method, particles are placed only at the interface positions within interfacial cells, and corrections are applied using a high-order kernel function. This approach improves accuracy while reducing computational costs compared to conventional methods. Figure 1 presents a schematic illustration of the PLS/KC method. This section provides a detailed explanation of the advection of the Level Set function, particle placement strategy, and correction method. In the following section, the performance of the PLS/KC method is evaluated through numerical experiments.

(1) Strategy for Advection of the Level Set Function

In conventional PLS methods, high-order advection schemes such as the WENO scheme and high-order TVD Runge-Kutta schemes have been employed to achieve accurate advection [2]. However, these methods are computationally expensive, posing a practical burden. The PLS/KC method adopts a strategy that simplifies the advection scheme while ensuring accuracy through particle-based correction. Specifically, a second-order ENO scheme and a second-order TVD Runge-Kutta scheme are used for advection. The ENO scheme effectively suppresses numerical oscillations near the interface while enabling stable computations at a relatively low cost.

In the PLS/KC method, the Level Set function is corrected using the positional information of particles placed on the interface. By appropriately utilizing particle information, the degradation in accuracy caused by advection calculations is compensated, ensuring interface shape preservation and mass conservation without the need for high-order advection schemes. This strategy enables the PLS/KC method to achieve sufficient accuracy while significantly reducing computational costs compared to conventional approaches.

(2) Particle Placement Strategy

In the conventional PLS method, particles are typically placed randomly within a three-cell range on both sides of the interface. The PLS/KC method follows this general approach but refines it by placing particles randomly only within interfacial cells and subsequently attracting them toward the interface. Similar to the PLS method, up to 64 particles are assigned per cell. However, by restricting the placement region to interfacial cells only, the computational cost is reduced while maintaining a particle distribution that effectively adapts to interface shape variations.

(3) Correction of the Level Set Function Using a High-Order Kernel Function

In the PLS method, Lagrangian particles are utilized to correct the Level Set function. In conventional PLS methods [4][5], linear interpolation has been used to compute the correction values. In contrast, the PLS/KC method improves correction accuracy by applying a high-order kernel function.

Specifically, a correction approach is introduced in which the Level Set function values at grid points are adjusted based on the positional information of interfacial particles.

The advection of particles is performed by interpolating the velocity at grid points and advancing the particle positions using a third-order Runge-Kutta scheme. Since the Level Set function should be zero at the particle positions on the interface, local corrections are applied based on particle positions, as formulated in Equations (6-7).

$$\varphi_{i,j,k}^{New} = \varphi_{i,j,k} - \lambda_{i,j,k} \tag{6}$$

$$\lambda_{i,j,k} = \frac{\sum w_{kernel}\,\varphi_P}{\sum w_{kernel}} \tag{7}$$

$$W_{spiky}(r,h) = \frac{15}{\pi h^6}\begin{cases}(h-r)^3 & 0 \le r \le h \\ 0 & otherwise\end{cases} \tag{8}$$

The correction using the kernel function is an interpolation method based on interfacial particle information, ensuring accuracy independent of the particle number density within the grid. Since the kernel function depends on the distance between particles and grid points, it facilitates straightforward extension to three-dimensional problems. In this study, the Spiky kernel function, adopted in the PLS/KC method, is employed, and its performance is evaluated through benchmark tests.

4. GPU-ACCELERATED FLOW SOLVER

The fluid solver is implemented using FORTRAN90 with OpenACC to enable GPU computing, thereby facilitating large-scale simulations. Programming with OpenACC allows straightforward GPU acceleration by simply adding directives, similar to OpenMP. In particular, if a multi-core CPU computing code based on OpenMP is already available, the effort required for GPU programming is significantly reduced. In this study, the OpenMP-based program code was replaced with OpenACC, enabling computations on a single GPU.

To evaluate the speedup of a single GPU relative to a 120-core CPU computation, Fig. 2 presents a performance comparison between a 120-core CPU and a single GPU. The test case used for this evaluation is a practical dam-break problem, which will be described in the next section. To examine the scalability and acceleration achieved through GPU computing, simulations were conducted for up to 1.5 billion grid points, and the corresponding speedup results for the 120-core CPU and single GPU are shown in Fig. 3. The results indicate that as the computational scale increases, the performance benefit of GPU acceleration becomes more pronounced. Even for grid sizes on the order of several million, the single GPU achieved more than five times the speedup compared to the 120-core CPU. The GPU used in this study is equipped with 80 GB of memory, allowing the proposed fluid solver to handle approximately 1.5 billion grid points on a single GPU.

As a limitation of this study, the computations of each term in the fluid solver are divided into subroutines, with OpenACC directives inserted using the DATA construct in each subroutine. This means that data transfers between the device and the host occur at the subroutine level. Although the fluid solver is fully GPU-accelerated, part of the interface capturing PLS/KC method operates using OpenMP. This is because the parallelization of the process for registering and managing PLS particles within computational cells is complex, and GPU acceleration has not yet been implemented for this part.

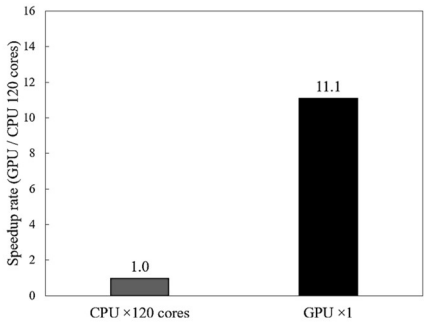

Fig. 2. Acceleration comparison of 120 CPU cores and single GPU

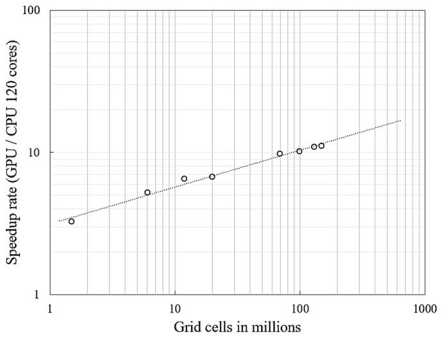

Fig. 3. Results of computational speedup of one GPU against 120 CPU cores (up to 150 million grid cells)

By optimizing these limitations, further acceleration is expected. This remains an issue for future research.

5. NUMERICAL TESTS

To evaluate the accuracy and mass conservation properties of the PLS/KC method as an interface capturing technique, two-dimensional and three-dimensional benchmark tests, which are commonly used in interface capturing schemes, were conducted. The purpose of these benchmark tests is to assess the accuracy of interface reconstruction.

(1) Two-Dimensional Single Vortex Test

The single vortex test is a two-dimensional benchmark test designed to evaluate large interface deformations caused by a time-dependent unsteady deformation velocity field [6]. The computational conditions follow [6], where a circle of radius 0.15 is placed within a unit square domain with an initial center coordinate of (0.5,0.75). A time-dependent velocity field is applied. The test duration is set to $T = 8.0$, and the velocity field is reversed at $t=T/2$, ensuring that the interface returns to its initial position and shape at $t=T$. The initial circular interface undergoes deformation and elongation due to the velocity field, forming a thin spiral shape at $t = T/2$. Subsequently, the reversed velocity field transports it back, restoring the initial configuration at $t=T$.

The computational domain is discretized using a uniformly spaced orthogonal grid, with grid resolutions of 32, 64, and 128 cells in each direction. The CFL number is fixed at 0.25 [6]. Fig.

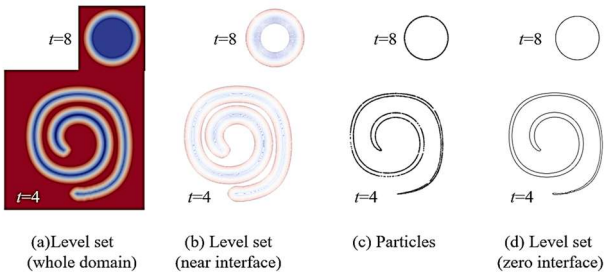

Fig. 4 Results of single vortex flow test at t=4 and 8 on 128x128 grid cells.

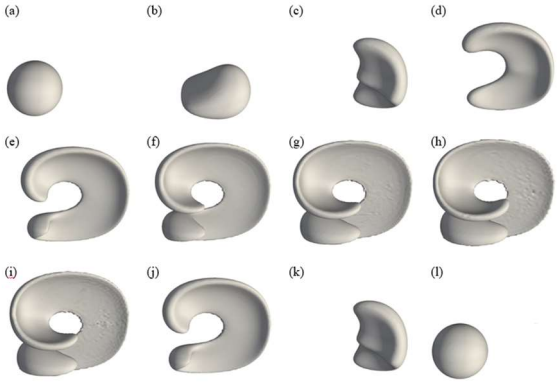

Fig. 5. Evolution of LeVeque's 3D deformation test with 100x100x100, N_{seed} =64 by PLS/KC method. (a)t=0.0, (b)t=0.2, (c)t=0.4, (d)t=0.6, (e)t=0.8, (f)t=1.0, (g)t=1.2, (h)t=1.4, (i)t=1.8, (j)t=2.2, (k)t=2.6, (l)t=3.0

4 illustrates the interface and particle distribution at $t=T/2$ and $t=T$. The PLS/KC method maintains a uniform particle distribution along the interface throughout the simulation, and it successfully preserves thin interfacial structures that would otherwise be lost in the conventional Level Set method.

(2) Three-Dimensional Deformation Test

LeVeque's three-dimensional deformation test [7] is a benchmark test used to analyze interfaces undergoing large deformations due to a time-dependent unsteady deformation velocity field. Since the interface undergoes significant stretching and compression, this test is widely employed to evaluate the mass conservation and shape-preserving properties of interface capturing methods. In this test, a sphere of radius 0.15 is placed within a unit cube domain, with an initial center coordinate of (0.35, 0.35, 0.35). A time-dependent unsteady deformation velocity field is applied. The test duration is set to T = 3.0, and the velocity field is reversed at $t = T/2$ so that the interface returns to its initial position and shape at $t = T$. The interface undergoes significant deformation due to the velocity field, reaching its most elongated shape at $t = T/2$. The reversed velocity field then transports it back, restoring the initial configuration at $t = T$.

Fig. 5 illustrates the interface and particle distribution from t = 0.0 to T. The PLS/KC method maintains a uniform particle distribution along the interface throughout the simulation and successfully preserves thin interfacial structures that would otherwise be lost in the conventional Level Set method.

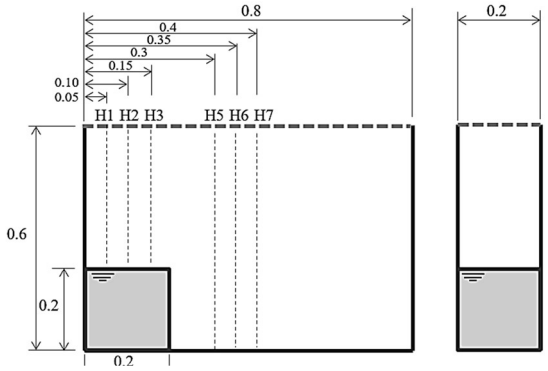

Fig. 6. Test conditions for the dam breaking benchmark by Nik (2017) [8].

(3) Practical Dam-Break Simulation

To evaluate whether the interface capturing method can accurately reproduce real physical phenomena and to verify its practical applicability, a three-dimensional dam-break test was conducted. This benchmark test is a representative example of free-surface flow analysis and has been extensively studied through both numerical simulations and experimental research [8]. In this test, (a) the position of the water front after dam collapse and (b) the water level variation were evaluated. The water-air interface was tracked using the improved Particle Level Set (PLS/KC) method proposed in this study. The interface capturing scheme based on the PLS/KC method was integrated into the fluid solver, and its applicability was assessed by comparing the simulation results with previous dam-break experiments [8]. As shown in Fig. 6, the experiment by Nik (2017) [8] was reproduced using a water tank of 0.8 m in length, 0.6 m in height, and 0.2 m in width. The initial water column measured 0.2 m × 0.2 m, and at the start of the simulation, a gate was vertically removed, allowing the water to flow out. The numerical simulation was performed under fluid conditions matching those in the experiment [8], with the air and water parameters listed in Table 1. The computational domain was discretized into a 640 × 160 × 480 uniform grid, resulting in a total of approximately 49 million grid cells.

Fig. 7 presents a snapshot of the simulation results from t = 0.0 to 0.8 seconds. The process of air entrainment caused by the dam break was captured by the free-surface shape, which represents the gas-liquid interface. It was observed that the interface was effectively tracked using the Level Set function and particles employed in the PLS/KC method. Approximately 0.8 million particles were used for correction. The Level Set function corrected by the PLS/KC method was applied to grid points located within five grid cells from the interface, corresponding to approximately 1.5 million grid cells out of the total 49 million in the computational domain.

The dimensionless displacement of the water front, defined as $x^* = x / h_0$, was recorded over time and is shown in Fig. 8. The numerical simulation results were compared with experimental data [8], the VOF method [9], and the Level Set method without particle correction. The results showed a slight deviation from the experimental values in the early stages of the simulation; however, the error magnitude was comparable to that of the VOF method [8]. In the later stages, the agreement with the experimental data improved.

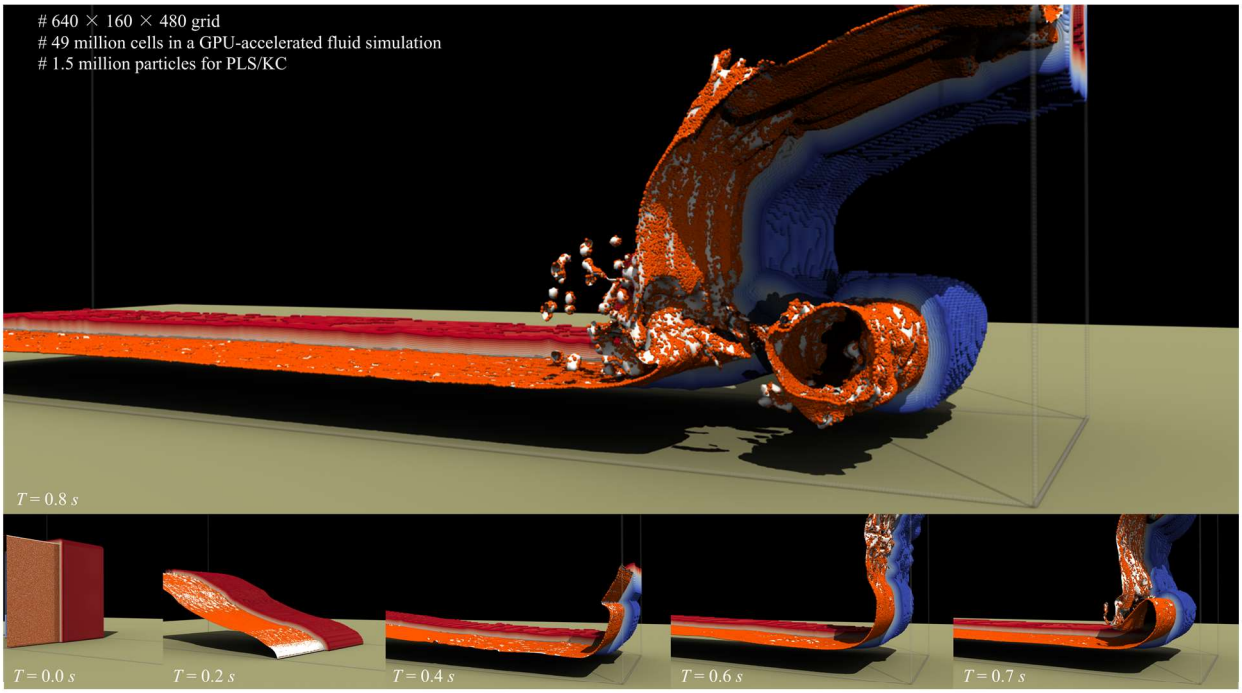

Fig. 7. A snapshot of the simulation results of dam breaking test from t=0.0s to 0.8s

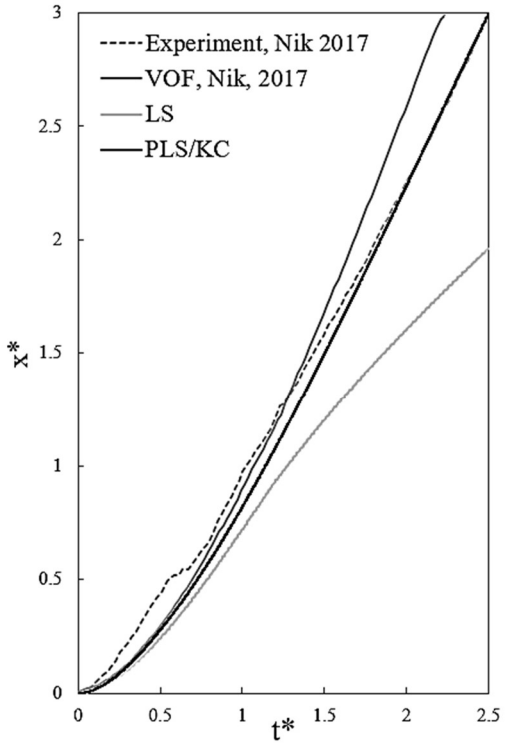

Fig. 8 Comparison of the surge front propagation after dam gate removal with the experimental data [8], the VOF calculation results [9], and the original LS method.

Table. 1. Physical properties at temperature of water and air 10 degC., Nik (2017) [8]

Physical properties	Water	Air
Density (kg/m^3)	999.7	1.2
Molecular viscosity (kg/(m s))	1.307×10^{-3}	1.4×10^{-5}
Surface tension coefficient (N/m)	0.0742	

6. SUMMARY AND CONCLUSION

In this study, an improved Particle Level Set method (PLS/KC) was integrated into a fluid solver, and a GPU-accelerated parallel computing environment was developed to achieve high-accuracy multiphase flow analysis and large-scale simulations. The PLS/KC method enhances interface accuracy and improves mass conservation by placing particles near the interface and applying corrections using a high-order kernel function. GPU optimization using FORTRAN90 + OpenACC enabled large-scale parallel computations, significantly improving computational efficiency. Numerical validation demonstrated that the interface capturing scheme using the PLS/KC method exhibited superior shape preservation and mass conservation compared to conventional methods in both two-dimensional and three-dimensional benchmark tests. Furthermore, GPU performance evaluations showed that the proposed approach achieved up to 11.1 times speedup compared to a 120-core CPU and was capable of handling large-scale simulations with 1.5 billion grid points.

Future challenges include further optimization of the PLS/KC method and full GPU implementation to achieve additional speedup. Moreover, advancing applied research on more complex physical phenomena is expected to further enhance the practicality of the PLS/KC method.

ACKNOWLEDGMENT

The computation was carried out using the computer resource *GENKAI* offered under the category of General Projects by Research Institute for Information Technology, Kyushu University.

REFERENCES

[1].Osher, S., & Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. Journal of Computational Physics, 79(1), 12-49.

[2].Enright, D., Fedkiw, R., Ferziger, J., & Mitchell, I. (2002). A hybrid particle level set method for improved interface capturing. Journal of Computational Physics, 183(1), 83-116.
[3].Kurioka, S., & Hu, C. (2025). Improved particle level set method with higher-order kernel function correction: Enhancing accuracy and conservation. Computers & Fluids, 291
[4].Mihalef, V., Metaxas, D., & Sussman, M. (2007). Textured liquids based on the Marker Level Set. Computer Graphics Forum, 26(3), 457-466.
[5].Zhao, L., Mu, K., Mao, J., Hongvan, K., & Peng, D. (2020). A three-dimensional one-layer particle level set method. International Journal of Numerical Methods for Heat & Fluid Flow, 30(7), 3653-3684.
[6].Rider, W. J., & Kothe, D. B. (1998). Reconstructing volume tracking. Journal of Computational Physics, 141(2), 112–152.
[7].LeVeque, R. J. (1996). High-resolution conservative algorithms for advection in incompressible flow. SIAM Journal on Numerical Analysis, 33(2), 627-665.
[8].Nik, M. (2017). Experimental and numerical study of free surface flow impact on a vertical cylinder (PhD thesis). Kyushu University.
[9].Körner, C., Thies, M., Hofmann, T., & Singer, R. F. (2005). Lattice Boltzmann model for free surface flow for modeling foaming. Journal of Statistical Physics, 121(1), 179–196.