Implementation of GPU-based Adaptive Particle Refinement Numerical Scheme for Smoothed Particle Hydrodynamics: Preliminary Test for FCI Jet Breakup Simulation

Do Hyun Kim [a], Hae Yoon Choi [a] , Eung Soo Kim [a*]
[a] *Department of Nuclear Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul, South Korea*
[*] *Corresponding author: kes7741@snu.ac.kr*

1. Introduction

Fuel Coolant Interaction(FCI) at light water reactor is an important phenomenon in nuclear safety analysis, and it includes variety of complex physical phenomena. At FCI premixing stage, the ejected molten corium jet first comes into contact with coolant and then breaks into small droplets, which is called jet breakup. Understanding the physics of jet breakup phenomenon plays an important role for more precise FCI analysis.

For several years, various techniques were developed to analyze the jet breakup phenomenon, using either Eulerian grid based methods or Lagrangian particle based methods[1]. Recently, meshless methods such as Smoothed Particle Hydrodynamics(SPH), which can effectively track multi-fluid interfaces due to its essential characteristics of the Lagrangian approach, have been widely used.

In this study, jet breakup phenomenon was simulated by SOPHIA code, which is a GPU-parallelized SPH solver developed by Seoul National University[2]. SOPHIA can effectively solve general fluid problems, but due to the nature of the explicit Lagrangian method, computational cost problems inevitably occur in high-resolution analysis. To reduce the computational cost and improve the local accuracy, Adaptive Particle Refinement(APR) model, which allows multi-resolution analysis by splitting the particles, was added to the SOPHIA code. For validation of the model, benchmark experiment was selected and results were compared with the simulation.

2. SPH Method

2.1 Concept of SPH method

SPH is a meshless Lagrangian method developed in astrophysical area[3], and it has been increasingly used for simulating fluid flows. The basic idea of SPH is to represent a fluid system as a set of finite number of particles with individual physical properties, such as mass, density, temperature and velocity. Each particle interacts with its neighboring particles to solve the governing equations that are discretized as SPH formulation. Thanks to its meshless nature, the SPH method has great advantages in handling free surfaces, flows with large deformations, and multiphase flows, compared to the mesh-based Eulerian methods.

2.2 SPH approximation

Mathematically, an arbitrary function can be expressed in integral form by using the delta function. The SPH method uses continuous kernel function to approximate the delta function, and expresses the equations in discretized form.

$$f_i(r) = \sum_j f_j W(r_i - r_j, h)\, V_j \qquad (1)$$

Subscript i and j denotes the central particle and the neighboring particles, respectively. V is the particle volume, and $W(r_i - r_j, h)$ stands for the kernel function, where h is the smoothing length representing the range of neighboring particles to be included in the approximation. In order to approximate the delta function, the kernel function must have a large value at the center and gradually decrease as it moves away from it, converging to zero. In addition, the various properties of the delta function must be satisfied. There are several types of kernel functions in SPH, such as Gaussian, Wendland, Quartic etc. In this study, Wendland2 kernel function was used. There are several forms of SPH gradient approximation, depending on its derivation. Following formula was adopted in this study.

$$\nabla f_i(r) = \rho_i \sum_j m_j \left(\frac{f_i + f_j}{\rho_i \rho_j}\right) \nabla W(r_i - r_j, h) \qquad (2)$$

2.3 SPH governing equation

In the SPH method, there are two ways to calculate density; mass summation and by solving the continuity equation. Mass summation was used in this study.

$$\rho_i(r) = \sum_j m_j W(r_i - r_j, h) \qquad (3)$$

The momentum conservation equation is described in Lagarangian form as equation (4).

$$\frac{d\boldsymbol{u}}{dt} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \boldsymbol{u} + \boldsymbol{g} \qquad (4)$$

where ρ and m are the density and mass of the fluid particle, and $\boldsymbol{u}$, p, and ν denote velocity, pressure, and kinematic viscosity, respectively. $\boldsymbol{g}$ is the gravitational constant. The SPH formulations of pressure and viscous term in equation (4) can be expressed by discretized form as equation (5) and (6).

$$\left(\frac{d\boldsymbol{u}}{dt}\right)_{fp,i} = -\sum_j m_j \left(\frac{p_i + p_j}{\rho_i \rho_j}\right) \nabla_i W_{ij} \qquad (5)$$

$$\left(\frac{d\boldsymbol{u}}{dt}\right)_{fv,i} = \sum_j \frac{4m_j\mu_j\overrightarrow{r_{ij}}\cdot\nabla_i W_{ij}}{(\rho_i+\rho_j)\left(|\overrightarrow{r_{ij}}|^2+\eta^2\right)}(\overrightarrow{u_i}-\overrightarrow{u_j}) \quad (6)$$

In the currently using SPH method, it is assumed that the fluids are weakly compressible(WCSPH), and the pressure is calculated by the Tait's equation below[4].

$$p = \frac{c_0^2\rho_0}{\gamma}\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right] \quad (7)$$

where ρ_0 denote the reference density of fluid, c_0 is the speed of sound, and γ is the polytropic index which is set to be 7.

SOPHIA was developed using CUDA C language, as a GPU parallelized code. Equations (1) through (7) are solved simultaneously on the thread assigned to each targeted fluid particle.

2.4 Adaptive particle refinement

In most SPH simulations, uniform particle distributions are used, regardless of scale or complexity of the problem. However, this can lead to an increase in computational cost by using too many particles for accurate simulation results, since higher resolution generally guarantees higher accuracy. To avoid this, multi-resolution scheme needs to be adopted, which allows us to use the appropriate number of particles as needed.

In this study, APR model suggested by Sun et al. [5] has been implemented into SOPHIA code, for enabling the multi-resolution capability. The APR is a method of using particles of different sizes within a simulation by splitting them according to given specific criteria. Fig. 1 shows the basic concept of the particle refinement. In two-dimensional model, the bigger(coarse resolution) particle is refined into 4 smaller(fine resolution) particles distributed on the vertices of a square.

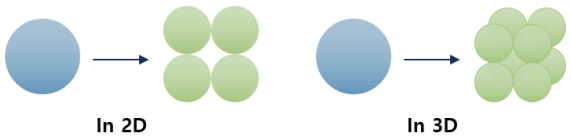

Fig. 1 Schematic of APR model

The relation between the variables of coarse particles and fine particles are written in 2D as follows.

$$\begin{aligned}
x_f^1 &= x_c - \frac{\sqrt{V_m}}{4},\ y_f^1 = y_c + \frac{\sqrt{V_m}}{4};\\
x_f^2 &= x_c + \frac{\sqrt{V_m}}{4},\ y_f^2 = y_c + \frac{\sqrt{V_m}}{4};\\
x_f^3 &= x_c - \frac{\sqrt{V_m}}{4},\ y_f^3 = y_c - \frac{\sqrt{V_m}}{4};\\
x_f^4 &= x_c + \frac{\sqrt{V_m}}{4},\ y_f^4 = y_c - \frac{\sqrt{V_m}}{4};\\
m_f &= \frac{m_c}{4};\quad V_f = \frac{V_c}{4};\quad h_f = \frac{h_c}{2};\\
\rho_f &= \rho_c;\quad p_f = p_c;\quad u_f = u_c;
\end{aligned} \quad (8)$$

where m, V and h are the mass, volume and smoothing length of the particle, ρ, p and u denote density, pressure and velocity, respectively. Subscript c and f represent coarse and fine particles, and superscripts 1 to 4 indicate the number of each fine particle that is refined. This process satisfies the mass, momentum and energy conservation.

The particle position-based criterion was used to trigger the refinement. Some of the particles are refined at the beginning of the simulation according to their position, and the rest of the particles are refined over time as the simulation progresses.

The APR process should be carried out sequentially, to avoid the conflicts between memories in the code[6]. To implement APR model in SOPHIA, CUDA-C atomic operation was introduced.

3. Simulation setup

In this study, the jet breakup phenomenon was simulated by SOPHIA, with APR model implemented. The experiment performed by Manickam et al. [7] was selected as the benchmark case for code validation.

3.1 Reference experiment

Fig. 2 shows the schematic of the MISTEE-Jet facility, which was built to perform the small scale jet breakup experiment. At the upper part of the facility, melt is prepared in crucible heated by induction furnace. During the heating, the plug covers the hole at the bottom side of the crucible. After the melt reaches the desired temperature, the plug is pulled out and the melt is ejected into a rectangular shape tank(465mm×75mm× 75mm) filled with water.

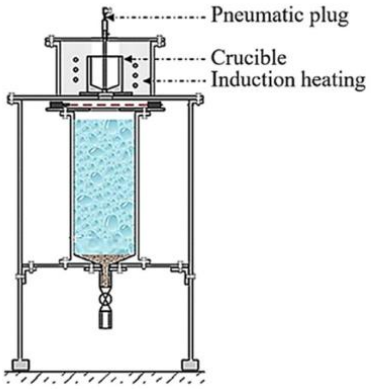

Fig. 2 Schematic of MISTEE-Jet facility

Woods metal, which has a melting point lower than the boiling point of water, was used to simulate the metallic component of corium. Since the initial temperature of the melt was set to be 70℃, boiling of water was not considered. The density of the Woods metal and water are $9730\text{kg}/m^3$ and $1000\text{kg}/m^3$, respectively. The initial jet velocity was set to be 1.7m/s

and the jet diameter is 10mm.

3.2 SPH simulation

2-dimensional jet breakup simulation was performed using the SOPHIA code. The initial condition and geometry are shown in Fig. 3.

To evaluate the performance of the APR model, simulations were performed for three cases with different resolutions: (Case 1) Fully refined case(w/o APR) with initial particle distance $\Delta x = 0.2mm$; (Case 2) Unrefined case(w/o APR) with $\Delta x = 0.4mm$; (Case 3) Partially refined case(APR) with $\Delta x = 0.4mm$.

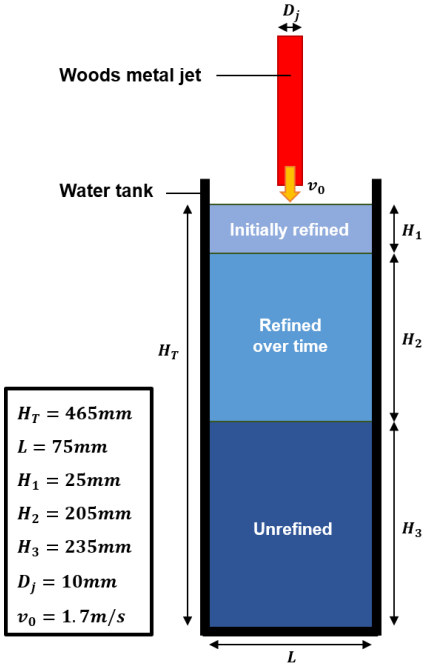

Fig. 3 SPH simulation geometry

In case 3, woods metal particles and water particles at the top of the tank are initially refined.(Fig. 3) As the simulation progresses, the melt jet heats the water surface and penetrates toward the interface between refined and unrefined area. When leading edge of the jet gets close to the interface, part of the particles in the unrefined area near the leading edge are sequentially refined to increase the resolution around the jet as shown in Fig. 4.

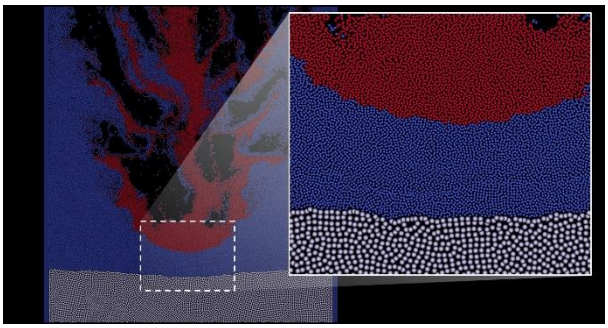

Fig. 4 Particle refinement at leading edge

For all cases, simulation time step was 10^{-6}s, and the sound speed was set to be 20m/s. Total simulation time is 0.2 sec.

4. Result and discussion

4.1 Simulation results

Fig. 5 shows the SPH simulation results of case 1, compared to the experimental result. When the melt jet penetrates into the water, the momentum difference between two liquid phases decelerates the jet, forming an inverted mushroom-like shape at the leading edge.

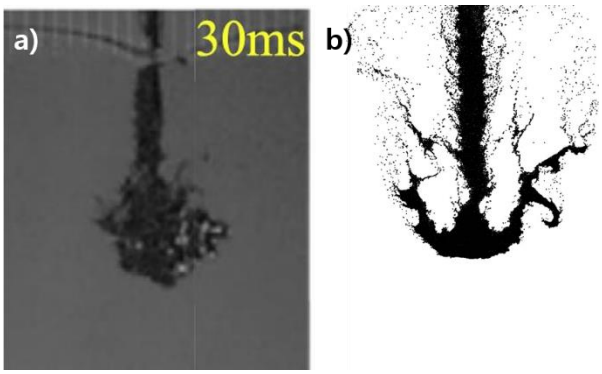

Fig .5 Mushroom-like shape at the leading edge of the jet
(a) Reference experiment (b) SPH simulation case 1

4.2 Evaluation on APR model

To validate that the APR model has been properly applied, the results of case 3 were compared with case 1, which is a high-resolution case.(Fig. 6) The shape of the jet and leading edge, and the penetration depth showed similar results. Compared to the low-resolution case(case 2), APR case precisely simulated the droplets that escaped from the jet. (Fig. 7)

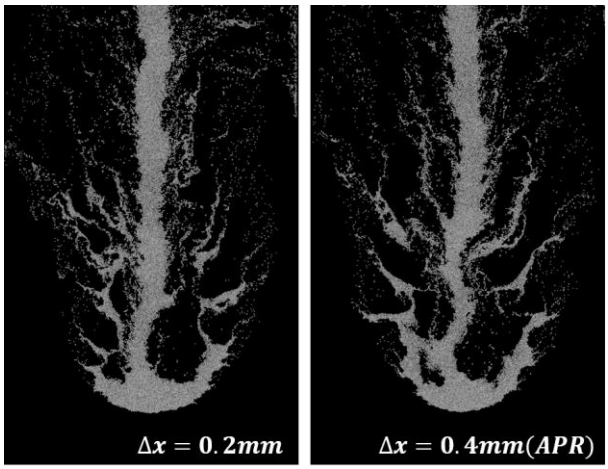

Fig. 6 Simulation results in different resolution
(a) Case 1 (b) Case 3

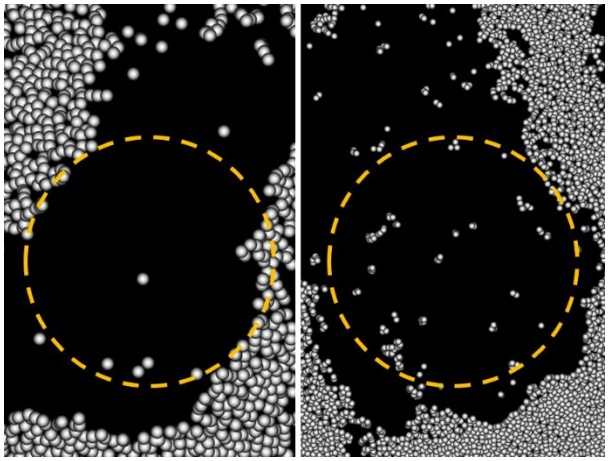
Fig. 7 Melt droplet escaped form the Jet: (a) Case 2 (b) Case 3

To evaluate the improvement in computing efficiency of the APR model, the total computing time was measured for each case. (Table I) Comparing case 1 and case 3, the simulation is performed at the same resolution in the region near the jet, but the computing time for case 3 is much less than that of case 1, at 62% level. This result suggests that APR model greatly contributes to the computational efficiency of the code, and the effect is expected to be maximized in 3D analysis, which requires much more particles and memory.

Table I. Computing time measurement on each case

	Particle distance [mm]	Number of particle [#]	Time [s]
Case 1	0.2	963,017	32855.9
Case 2	0.4	245,059	8349.6
Case 3	0.4 → 0.2	245,056 → 646,564	20455.0

5. Summary

In this study, the jet breakup phenomenon was simulated by the SPH method. To implement the multi-resolution scheme, adaptive particle refinement(APR) model was adopted in the in-house SPH code SOPHIA. The result of the simulation shows good agreement with the benchmark experimental data in qualitative manner. The newly adopted model, APR was evaluated in the terms of computing efficiency, and it was confirmed that the computing time could be reduced to about 62% compared to the previous method.

ACKNOWLEDGEMENT

This research was supported by Nuclear Energy Technology Development Program(U.S.-ROK I-NERI Program) through the National Research Foundation of Korea(NRF) funded by the Ministry of Science and ICT(NRF-2019M2A8A1000630)

REFERENCES

[1] S. Park, H. S. Park, B. I. Jang & H. J. Kim (2016). 3-D simulation of plunging jet penetration into a denser liquid pool by the RD-MPS method, Nuclear Engineering and Design, Vol. 299, 154-162.
[2] Y. B. Jo, S.H. Park, H. Y. Choi, H. W. Jung, Y. J. Kim & E. S. Kim (2019). SOPHIA: Development of Lagrangian-based CFD code for nuclear thermal-hydraulics and safety applications, Annals of Nuclear Energy, Vol. 124, 132-149.
[3] R. A. Gingold, J. J. Monaghan (1977). Smoothed Particle Hydrodynamics: Theory and Application to Non-Spherical Stars, Monthly Notices of the Royal Astronomical Society, Vol. 181, No.3, 375-389.
[4] J. J. Monaghan (1994). Simulating free surface flows with SPH, Journal of computational physics, Vol. 110, No.2, 399-406.
[5] P. N. Sun, D. Le Touzé, G. Oger, A. M. Zhang (2019). Derivation and validation of a δ-SPH model for simulating strongly-compressible multiphase flows, The 14th SPHERIC International Workshop, 84-91.
[6] Q. Xiong, B. Li & J. Xu (2013). GPU-accelerated adaptive particle splitting and merging in SPH, Computer Physics Communications, Vol. 184, 1701-1707.
[7] L. Manickam, S. Bechta & W. Ma (2017). On the fragmentation characteristics of melt jets quenched in water, International Journal of Multiphase Flow, Vol. 91, 262-275.