Status of Particle-based Multi-physics Simulation for Nuclear Severe Accident: Review

Eung Soo Kim [a*], So-Hyun Park [a], Young Beom Jo [a]

[a]*Dept. of Nuclear Engineering, Seoul Nat'l. Univ., 1 Gwanak-ro, Gwanak-gu, Seoul, Korea*

[*]*Corresponding author: kes7741@snu.ac.kr*

1. Introduction

After the accident at Fukushima Dai-Ichi nuclear power plants, it was revealed that revealed that some accidents, within very low probability, may result in severe accidents accompanying core melting, plant damage, and release of radioactive materials to environment. In this regard, every county became to require significant safety enhancements to the conventional nuclear power plants with updating evaluations on the potential impact from seismic and flooding events, adopting new equipment to better handle potential core damage events, etc. [1].

After Fukushima accident, one notable challenge is that the people require to handle much more complicated multi-physics phenomena than the traditional design-basis-accidents (DBAs) which required only thermal-hydraulics knowledge. Therefore, to address this type of issues properly, first of all, we have to clearly identify, understand and predict the key phenomena associated with the progress of the accident [2].

In the nuclear society, analyses of nuclear reactor safety have been based on the empirical models obtained from extensive experimental data. Although this type of analysis can run quickly, its critical drawback is that the use is only limited to the systems and conditions for which the experiments are conducted. Especially, these approaches are often challenging for the severe accident issues, because they would be too expensive, too dangerous, or even impossible to study by direct experimentation.

In recent years, many scientists and engineers are making efforts to advance technology to find new ways to tackle nuclear systems & safety challenges—challenges that cannot be addressed through traditional method alone. These methods are mostly based on the 1st principle physics with multi-dimensional, multi-scales, and multi-physics. In addition, with the advanced computational modeling and predictive simulation capabilities, high performance computing (HPC) reduces the risks and uncertainty that are often barriers to industry adopting new and innovative technologies.

2. SOPHIA Code

SOPHIA code is an open-source particle-based multi-physics code developed in Seoul National University (SNU) for nuclear thermal-hydraulics & safety applications. This code is based on the particle-based numerical schemes such as smoothed particle hydrodynamics (SPH), discrete element method (DEM), and etc. Unlike the conventional grid-based methods (CFD, FEM, and etc), the particle-based method divides a domain of interest into small particles (or parcels) and solves governing equations by tracing their positions. Because of this Lagrangian nature, it can handle large deformation without interface tracking and incorporate various fluids, phases, and physics easily in the models together.

In the SOPHIA, the governing equations and the physical models are solved by kernel interpolation or direction interactions over the neighboring particles. The SOPHIA code is currently under development to handle the following phenomena: (1) fluid flow, (2) heat transfer, (3) turbulence, (3) melting/solidification, (4) multi-phase, (5) multi-fluids, (6) multi-components, (7) diffusion, (8) natural convection, (9) solid granular flow, (10) chemical reaction, (11) rigid body, (12) elastic deformation, etc [26-29]. However, intrinsically, it can be easily extended to and become well suitable for many other phenomena with slight necessary modifications. The SOPHIA code is written in C++ and parallelized using Graphical Processing Units (GPUs). To improve the computational performance, various parallelization schemes, including parallel mapping, reduction, and parallel sorting are adopted. This open-source code is now available at OECD-NEA Databank and distributed to the researchers and the engineers for non-commercial use [4].

3. Basic Theory

The SOPHIA codes is based on two particle-based numerical methods; SPH and DEM. The basic concepts on these methods are as follows.

The SPH is based on an interpolation method which is basically the theory of integral interpolants using delta function.

$$f(x) = \int_{\Omega} f(x')\delta(x - x')dx' \qquad (1)$$

The variable x denotes the point in volume Ω, and δ denotes the Dirac delta function which has a value of zero everywhere except for at a certain point and whose integral over the entire region is equal to one. The discretized formulation of SPH particles can be obtained by using the kernel functions that approximate a delta function.

$$f(r_i) = \sum_j f(r_j) W(r_i - r_j, h) V_j \tag{2}$$

The variable $f(r_i)$ is a function at the position i, subscript j represents the adjacent particles of particle i, V is the particle volume, and $W(r_i - r_j, h)$ stands for the kernel function, where h denotes influencing area of the kernel weighting function. The kernel function is a symmetric weighting function of particle distance which should be normalized over its support domain. The simplified principle of SPH approximation is described in Fig. 1.

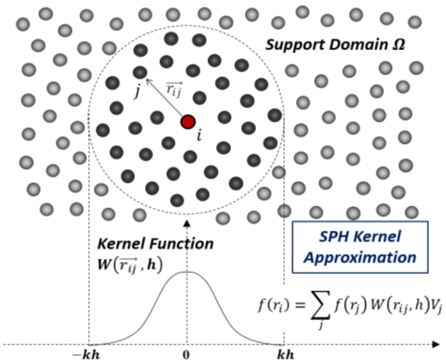

Fig. 1. SPH Particle kernel approximation [5]

The main governing equations describing the physics can be formulated using the above SPH interpolations. The details can be found in many previous literatures [4-6]. In the SOPHIA code, most of the physical/chemical processes are solved using the SPH method except for the discrete solid behaviors.

For modeling solid particulate behaviors, the SOPHIA code uses Discrete Element Method (DEM). In the DEM method, the collision and following transition and rotation of each rigid body are directly solved in detail. In this method, the collision between rigid bodies is generally solved using spring-dashpot model based on the soft-sphere approach as shown in Fig. 2.

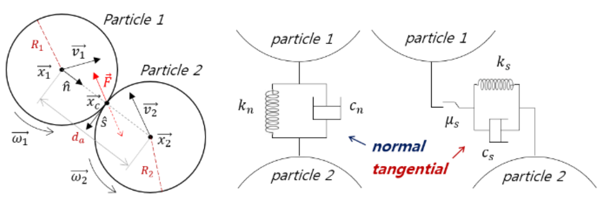

Fig. 2. Soft-sphere spring-dashpot collision model [5]

Based on the above spring-dashpot collision model, the contact force between two particles in normal and tangential direction consists of elastic term and damping term while the tangential force is determined through an additional comparison with the frictional force.

$$\vec{f_c} = (k_n \delta_n - c_n |\vec{v_{cn}}|)\hat{n} + f_s\,\hat{s} \tag{3}$$

$$f_s = \min\left(k_s \delta_s - c_s |\vec{v_{cs}}|\,,\, f_{friction}\right) \tag{4}$$

The subscript n and s represents the normal and tangential direction in the collision, and the variable k and c denotes the spring stiffness and damping coefficient between two rigid bodies. Also, the variable δ stands for the overlap between particles at the moment of collision, while $\vec{v_c}$ is the relative velocity vector between particles at the contact point. In the SOPHIA code, the Hertz-Mindlin contact force model is applied for the analysis of particulate system [5].

In the SOPHIA code, the SPH and the DEM models are hydraulically coupled together. To achieve this, unresolved coupling approach which allows the overlap between SPH-DEM particles is adopted considering the size of the debris particles as shown in Fig. 3. The fluid equations for the SPH particles are solved based on the local porosity within coupling domain, and the dominant forces acting on the DEM solid particles are estimated from the information of the adjacent SPH particles. In addition, the interaction force acting on the SPH particles is also calculated by the Newton's third law of motion.

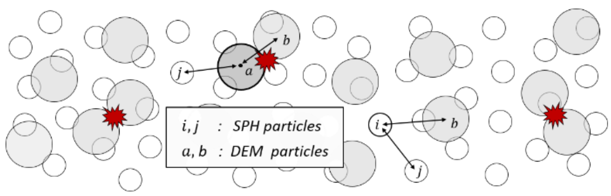

Fig. 3. Concept of SPH-DEM unresolved coupling [5]

4. Parallelization

Intrinsically, the particle-based methods require a large computational cost, which was the biggest technical barrier for practical engineering-scale simulations. However, the recent advancement in high-performance computing such as a graphical process unit (GPU) enables to overcome this issue. A GPU is a specialized processor originally designed to accelerate the graphics of 3-D rendering. The GPU consists of a number of streaming multiprocessors (SM), which are made up of dozens of CUDA cores again. GPUs can process many pieces of data simultaneously, and that is why GPU is becoming popular in a wide range of applications including artificial intelligence (AI). In this section, the basic parallelization concept in the SOPHIA code is briefly summarized and the details can be referred to the literature [6].

In the SOPHIA code, the parallelization can be achieved by two-levels. At the upper level, multiple GPUs execute individual main loops for each subdomain concurrently, and at the lower level, each GPU splits the main loop into two streams (Fig. 4). In one stream, computation is conducted in the other stream, data transfer is done simultaneously. The overall multi-GPU parallelization process is as follows.

(1) Computational domain is decomposed into multiple subdomains (**Multi-GPU Allocation**).
(2) Each subdomain is partitioned into outer particle region and inner particle region (**Sub-domain Division**).
(3) Outer particle regions of every GPU are computed (**Multi-GPU Concurrent Execution**).
(4) Inner particle regions of every GPU are computed (**Multi-GPU Concurrent Execution**).
(5) Concurrently, data exchange between adjacent GPUs is conducted (**Inter-GPU Communication**).

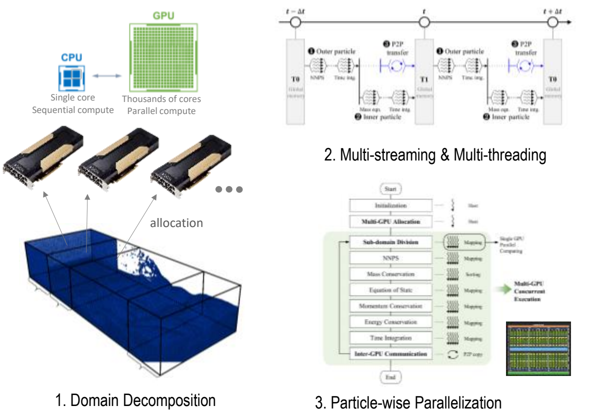

Fig. 4. Multi-GPU parallel concept for SOPHIA

After subdomains are assigned to GPUs, each GPU independently calculates the particles' behaviors. Nearest Neighbor Particle Search (NNPS) is the first step to be conducted. The NNPS is commonly optimized using an algorithm based on a uniform grid and a parallel sorting. The subsequent processes are performed during the SPH interpolation or the DEM calculation using these starting particle indices and ending particle indices [6]. Performance evaluation of this parallel algorithm was performed and reported in the previous literature [6], and as a result, it was demonstrated that the calculation speed can be increased by up to 2 orders of magnitude compared to the single CPU code as the number of particles increases up to 1 million particles [6].

5. Nuclear Severe Accident Applications

Applications of the SOPHIA code to the nuclear severe accidents have been successfully conducted in several previous researches, and this paper introduces some of notable activities [5-8].

Fuel Coolant Interaction (FCI) is an phenomenon in the LWR severe accident, which occurs when molten fuel falls into coolant or vice versa. In addition, the interactions between coolant and molten fuel at high temperature involve many complicated physical phenomena such as hydraulic fragmentation, phase change, chemical reaction, explosion, etc. Nevertheless, conventional grid-based CFDs experience difficulties in modeling this phenomenon due to the complex physics and large deformation.

Recently, Park et. al. conducted simulation on this phenomenon using multi-phase & multi-fluid model. In this simulation, a jet column was injected into a water pool in a rectangular tank with 0.1 m in width and 0.02 m in depth, and 0.4 m in height. The jet diameter was 10 mm and the injection speed was 3.8 m/s. As shown in Fig. 5, the simulation reproduced the experiments very realistically [6]. Quantitative comparisons between the simulation and the experiment were also made and the agreement was reported to be great [6].

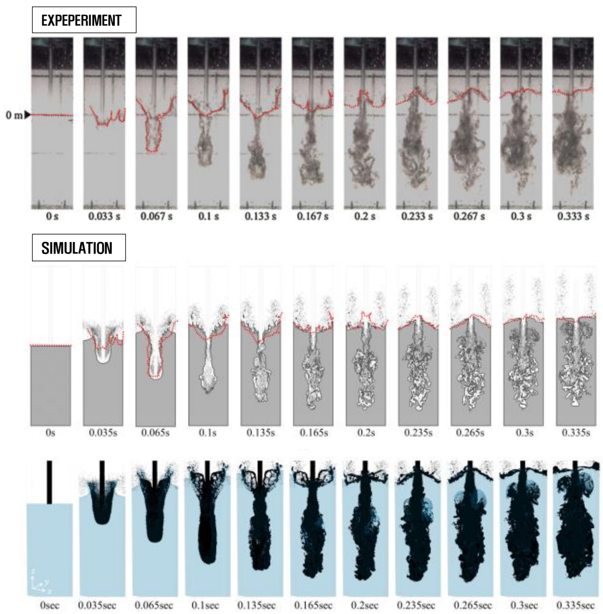

Fig. 5. FCI simulation using SOPHIA code [6]

Recently, the SOPIHA code was used for simulation of debris self-leveling phenomena in molten-corium-concrete-interaction (MCCI). In order to solve this problem, Jo [8] coupled the DEM with the multi-phase SPH, which can completely resolve the interface between liquid and vapor without empirical correlation. In this integration, the coolant and the vapor phases were solved by the SPH and the solid debris was solved by the DEM. Complete 2-way coupling between the SPH and the DEM was achieved for momentum exchange. The simulation was compared with the benchmark experimental data performed in Kyushu University [9]. Fig. 6 shows the simulation results with the experimental data. This study demonstrated that the SOPHIA code can reproduce the three-phase self-leveling phenomena in both qualitatively and quantitatively.

One recent applications of the SOPHIA code to the nuclear severe accident is the IVR-ERVC strategy [7]. In this study, the in-vessel corium behavior was modeled using SOPHIA code and the external-vessel natural circulation phenomena was modeled using MARS code. The SOPHIA-MARS code coupling is implemented using socket programming. The exchanged data are the heat flux and the temperature of the outer vessel wall. In the in-vessel corium simulation, natural circulation, phase separation, turbulence, thermal ablation, and crust formation were modeled. Fig. 7 shows a snapshot of the on-going full-scale demonstration analysis.

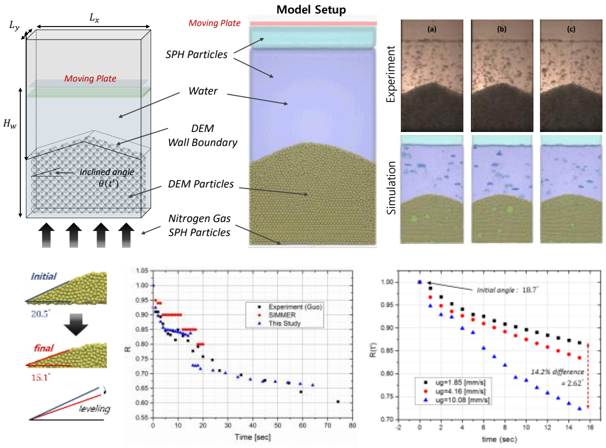

Fig. 6. FCI simulation using SOPHIA code [8]

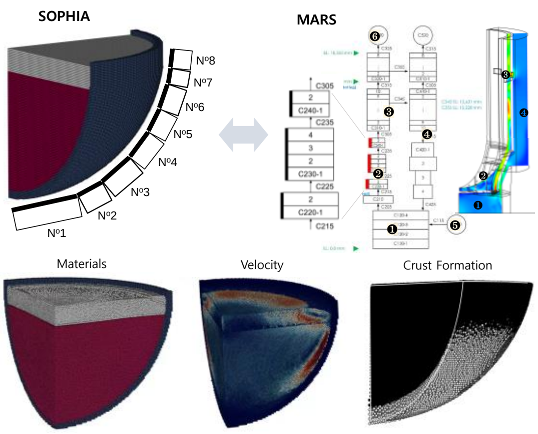

Fig. 7. FCI simulation using SOPHIA code [7]

4. Conclusions

Particle-based method is one of the emerging advanced modeling & simulation technologies in many science & engineering fields. The main advantages of particle-based method are the capability to handle large deformation without interface tracking and to incorporate various physics in convenience. Especially, over the past few decades, particle-based methods have exponential growth with the support of CPU and GPU hardware development. And it enables them to overcome their large computational cost, which has been considered as the biggest drawback.

This paper introduced the recent status of the SOPHIA code under development for nuclear safety applications. The latest SOPHIA code is based on the SPH and the DEM methods which are the most widely used particle-based numerical methods. In this code, the domain of interest is represented by a finite number of particles that carry individual properties and variables by solving discretized equations over the adjacent particles. In the course of several benchmark studies, the SOPHIA code exhibited great capability and potential for dealing with various complex severe accident phenomena that traditional grid-based methods have intrinsic difficulties in managing.

REFERENCES

[1] Saghafi, Mahdi, and Mohammad B. Ghofrani. "Accident management support tools in nuclear power plants: a post-Fukushima review." Progress in nuclear energy 92 (2016): 1-14.

[2] Trambauer, K., et al. Sarnet lecture notes on nuclear reactor severe accident phenomenology. No. CEA-R--6194. CEA Cadarache, 2008.

[3] Li, Gen, et al. "A review on MPS method developments and applications in nuclear engineering." Computer Methods in Applied Mechanics and Engineering 367 (2020): 113166.

[4] OECD/NEA Databank (https://www.oecd-nea.org/tools/abstract/detail/nea-1911/)

[5] Jo, Y.B., Park, S.H., Park, J.R., Kim, E.S., "Numerical Simulation of 3-Phase Debris Bed Hydrodynamic Behavior Using Multi-phase SPH-DEM Coupling, KNS Spring, 2021.

[6] Park, S.H., Jo, Y.B., Kim, E.S.*, Development of Multi-GPU-Based Smoothed Particle Hydrodynamics Code for Nuclear Thermal-hydraulics and Safety: Potential and Challenges, Frontiers in Energy Research, Vol. 8, 2020.

[7] Park, S.H., "Development of Lagrangian Particle-Based Methodology for In-Vessel Retention and External Reactor Vessel Cooling Application" Ph.D. Thesis, Seoul National University, 2021.

[8] Jo, Y.B. "Development of GPU-Based SPH-DEM Code for Hydrodynamic Interactions between Multi-Phase Flow and Solid Debris." Ph.D. Thesis, Seoul National University, 2020.

[9] Cheng, Songbai, et al. "Experimental studies and empirical models for the transient self-leveling behavior in debris bed." Journal of nuclear science and technology 48.10, 2011.