

GPU-Accelerated 3D Simulation of Molten Salt Spreading Using Moving Particle Method

Wiktor Gronek and Han-Young Yoon*

Department of Nuclear Power Plant Engineering, KEPCO International Nuclear Graduate School (KINGS)

*Corresponding author: hyoon@kings.ac.kr

***Keywords : Molten salt, Particle method, MSR safety, FLiNaK, GPU**

1. Introduction

There has been a renewed global interest in Molten Salt Reactors (MSRs) in recent years. The number of activities related to their design and technology is growing [1]. Commercialization of MSRs will require regulatory approval, which means that safety must be demonstrated through experiments and computational analysis, including severe accidents [2]. One such scenario assumes leakage of molten salt and exposure to outdoor conditions [3]. Molten salt solidifies upon contact with lower temperature surfaces. Understanding the solidification rate, and how far the material spreads before fully solidifying, is needed to design optimal containment systems.

Due to the free surface nature of the problem, a meshless Lagrangian approach can be used for its simulation [4]. In this approach, the fluid domain is divided into moving particles. Each particle carries variables such as velocity, pressure and temperature. This method has an advantage over the conventional Eulerian approach, as it avoids solving the advection term of the momentum equation. It also conserves mass inherently, since each particle retains constant mass throughout the simulation.

One widely used Lagrangian technique is the Moving Particle Semi-implicit (MPS) method. It was originally developed for incompressible free surface flows [5]. MPS has found application in nuclear engineering, mostly for simulating corium spreading after a reactor meltdown [6-9]. Its main drawback, however, is computational cost. Large-scale simulations require solving a large, sparse pressure matrix. This limits its practicality for full-scale, high-resolution 3D cases.

To overcome this limitation, we adopted a fully explicit variant of MPS, known as E-MPS [10]. Rather than solving a pressure matrix, E-MPS treats pressure as a state variable, which is computed using an artificial speed of sound [11]. Fully explicit approach is naturally well suited to GPU computation. We further extended the method with solidification and surface tension models, to capture the spreading and solidification behavior of molten salt.

In this study, we performed a 3D simulation of FLiNaK being poured onto a flat stainless-steel plate. We compared the results against two references: the 1D transient MELTSPREAD code developed by Argonne

National Laboratory [12], and a 2D analysis using the Volume of Fluid formulation in Ansys Fluent [13].

2. Numerical Methodology

The following numerical procedure is used in this study. In this process, 3D incompressible flow is assumed.

2.1. Governing Equations

The governing equations are the continuity, momentum, and energy equations:

$$\frac{D\rho}{Dt} = 0 \quad (1)$$

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho}\nabla P + \nu\nabla^2\mathbf{u} + \mathbf{g} + \frac{\mathbf{F}_s}{\rho} \quad (2)$$

$$\frac{DT}{Dt} = \frac{k}{\rho C_p}\nabla^2 T + Q, \quad (3)$$

where t is time, ρ the density, $\mathbf{u}$ the velocity vector, P the pressure, ν the kinetic viscosity, $\mathbf{g}$ the gravity, $\mathbf{F}_s$ the surface tension force, k the thermal conductivity, T the temperature, C_p the specific heat capacity, and Q the heat source.

2.2. Discretization for E-MPS method

In the E-MPS method, the spatial derivatives are evaluated through a particle-interaction model based on weighting functions. Two weighting functions are used: $w_{grad}(r)$ for the pressure gradient term and $w(r)$ for all other terms [10], defined as:

$$w_{grad}(r) = \frac{r_e}{r} - \frac{r}{r_e} \quad (4)$$

$$w(r) = \frac{r_e}{r} + \frac{r}{r_e} - 2 \quad (5)$$

where r_e denotes the effective radius of the interaction zone and $r = |\mathbf{r}_i - \mathbf{r}_j|$ is the distance between particles i and j , whose position vectors are $\mathbf{r}_i$ and $\mathbf{r}_j$. Both weighting functions vanish outside the effective radius ($r > r_e$).

The sum of the weighting function over the neighbours of particle i defines the particle number density n_i :

$$n_i = \sum_{j \neq i} w(|\mathbf{r}_i - \mathbf{r}_j|) \quad (6)$$

The Laplacian operator acting on a physical quantity φ are then discretised as:

$$\nabla^2 \varphi_i = \frac{2d}{n_0 \lambda} \sum_{j \neq i} [(\varphi_j - \varphi_i) w(|\mathbf{r}_i - \mathbf{r}_j|)] \quad (7)$$

where d is the number of spatial dimensions and the subscript 0 denotes a value evaluated for the uniformly arranged particle configuration with initial distance between particles l_0 . The coefficient λ is:

$$\lambda = \frac{1}{n_0} \sum_{j \neq i} |\mathbf{r}_i - \mathbf{r}_j|^2 w(|\mathbf{r}_i - \mathbf{r}_j|) \quad (8)$$

The pressure is obtained directly from the deviation of the particle number density through:

$$P_i = a^2 \frac{\rho^0}{n_0} (n_i - n_0) \quad (9)$$

In this E-MPS [10] method the following formulation is used for the discretized pressure gradient:

$$\nabla P_i = \frac{2d}{n_{0grad}} \sum_{j \neq i} \left[\frac{P_{ij}}{|\mathbf{r}_j - \mathbf{r}_i|^2} (\mathbf{r}_j - \mathbf{r}_i) w_{grad}(|\mathbf{r}_j - \mathbf{r}_i|) \right] \quad (10)$$

where

$$P_{ij} = \frac{P_i + P_j}{2} \quad (11)$$

where a denotes the artificial sound speed.

There is a restriction that the virtual Mach number $\frac{|u|}{a}$ should not exceed 0.2. Courant numbers regarding the fluid velocity and the virtual sound speed satisfy the following conditions [10]:

$$C_u \equiv \frac{u_{max} \Delta t}{l_0} < 0.2; \quad C_s \equiv \frac{a \Delta t}{l_0} < 1.0 \quad (12)$$

where u_{max} is the maximum absolute velocity among all particles and Δt needs to satisfy the above conditions.

2.3. Heat Transfer and Phase Change Model

Model assumes conductive heat transfer between fluid, solid and wall particles. The interaction between particles of different phases or materials is handled through the contact temperature T_c [9], defined as:

$$T_c = \frac{\sqrt{\rho_i C_{pi} k_i} T_i^n + \sqrt{\rho_j C_{pj} k_j} T_j^n}{\sqrt{\rho_i C_{pi} k_i} + \sqrt{\rho_j C_{pj} k_j}} \quad (13)$$

Phase change is simulated by two types of particles, fluid and solid particles, differing in their thermal energies per unit volume [9]. Temperature is the function of the thermal energy per unit volume as follows:

$$T = \begin{cases} T_s + \frac{H - H_{s0}}{C_{ps}} & (H < H_{s0}) \\ T_s & (H_{s0} \leq H \leq H_{s1}) \\ T_s + \frac{H - H_{s1}}{C_{pl}} & (H_{s1} < H) \end{cases} \quad (14)$$

where T_s is the melting temperature, H_{s0} , H_{s1} are the enthalpies at the start and at the end of the melting (difference between the H_{s0} and H_{s1} is latent heat L_{ls}), C_{ps} , C_{pl} are the specific heat of the solid phase and the liquid phase. The solid fraction β [9] is defined as:

$$\beta = \frac{H_{s1} - H}{H_{s1} - H_{s0}} \quad (15)$$

In this work, the viscosity of a mixture particle increases exponentially [14]:

$$\nu_{pc} = \nu_0 \cdot \exp(\beta \cdot 2.5) \quad (16)$$

After $\beta = 1$, particles are treated as immobile (fixed in position).

Fig. 1 shows the results of energy equation calculation including conductive heat transfer and phase change model. The physical properties for both solid and liquid: $k = 4.0 \times 10^2$ W/(m · K), $\rho = 1.0 \times 10^3$ kg/m³, $C = 1.0 \times 10^3$ J/(kg · K), $L_{ls} = 2.0 \times 10^5$ J/m³, $T_{wall} = -100^\circ\text{C}$, $T_s = 0^\circ\text{C}$, $T_{fluid} = 10^\circ\text{C}$. The results show strong agreement with the analytical solution [9], as shown in Fig. 2.

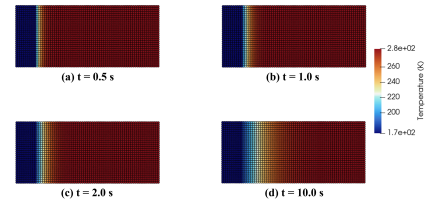


Fig. 1. Temperature contour maps during solidification.

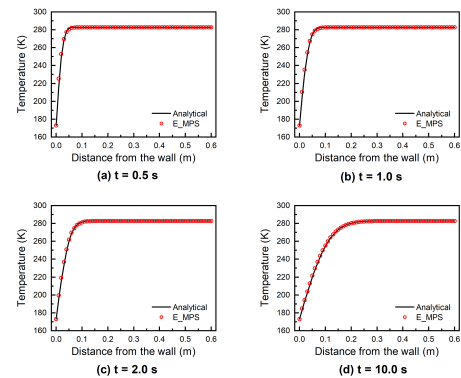


Fig. 2. Temperature profiles along the distance from the wall, comparing the E-MPS results with the analytical solution.

Additionally, the model assumes radiative heat transfer to the atmosphere, which acts only on surface particles:

$$Q_{rad} = \sum \varepsilon \cdot \sigma \cdot A_i \cdot (T_i^4 - T_\infty^4) \quad (17)$$

where ε is emissivity, σ is the Stefan-Boltzmann constant and A_i is the area covered by i -th particle.

2.4. Surface Tension Model

For the surface tension calculation, the potential model is used [15]. The force of surface tension $\mathbf{F}_s$ acting on particle i is defined as follows by using the potential function $\Gamma(r_{ij})$:

$$\mathbf{F}_s = - \sum_{j \neq i} \frac{\partial \Gamma(r_{ij})}{\partial r} \frac{\mathbf{r}_{ij}}{r_{ij}}. \quad (18)$$

The potential function $\Gamma(r_{ij})$ is defined as follows [15]:

$$\Gamma(r_{ij}) = C\gamma(r_{ij}) \quad (19)$$

where C is the coefficient of the potential function, is calculated before simulation and depends on surface tension coefficient.

$$\gamma(r_{ij}) = \begin{cases} \frac{1}{3}(r_{ij} - \frac{3}{2}l_0 + \frac{1}{2}r_e)(r_{ij} - r_e)^2 & (r_{ij} < r_e) \\ 0 & (r_{ij} \geq r_e), \end{cases} \quad (20)$$

The potential force acts as an attractive force while $r_{ij} > l_0$ and a repulsive force while $r_{ij} < l_0$. Initially, the cube-shaped water droplet became round due to the action of surface tension force, as shown in Fig. 3. After initial oscillations, diameter of the droplet becomes stable (Fig. 4).

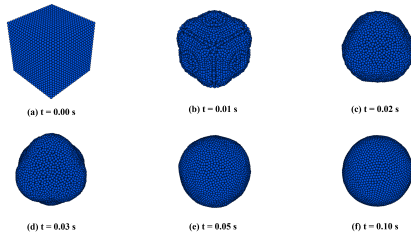


Fig. 3. Shape evolution of an initially cubic water droplet into a sphere under surface tension.

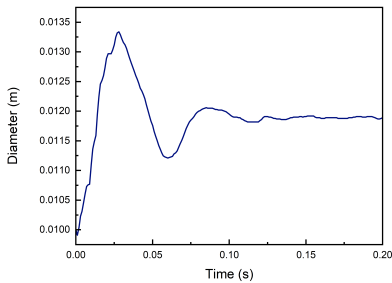


Fig. 4. Time history of the water droplet diameter.

Interaction between the fluid and the solid surface is calculated as follows:

$$\frac{C_{fs}}{C_{ff}} = \frac{1}{2}(1 + \cos \theta) \quad (21)$$

C_{fs} and C_{ff} are coefficient used for interactions between fluid-solid and fluid-fluid particles respectively, θ is contact angle. Fig. 5 shows the droplet shapes after stabilization for different contact angles between water and the surface.

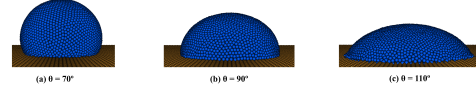


Fig. 5. Equilibrium water droplet shapes for different contact angles.

2.5. Algorithm of Calculations and GPU Implementation

The implemented algorithm is fully explicit and the overall calculation flow is shown in Fig. 6.

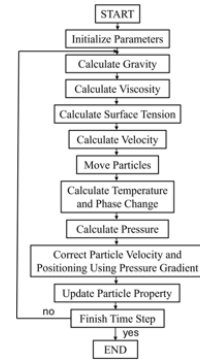


Fig. 6. Flowchart of the E-MPS calculation procedure.

The solver is implemented in CUDA C++, with particle fields stored as structure-of-arrays in CUDA Unified Memory and each physical term computed by a one-thread-per-particle kernel (256 threads per block). Neighbor search uses a uniform background grid with cell size r_e , limiting the search to the surrounding $3 \times 3 \times 3$ cells [16]. The grid is rebuilt each step via particle hashing, sorting by cell index (`thrust::sort_by_key`), and a boundary kernel that records the particle range of each cell.

3. Simulation Set-up

The modeled spill scenario involved eutectic LiF-NaF-KF (46.5-11.5-42 mol%), commonly referred to as FLiNaK, impinging on a flat stainless-steel substrate. A schematic illustration of the experiment is shown in Fig. 7. The geometry, initial conditions and material properties of the simulation are given in Tables I-II and are the same as those in the referenced MELTSPREAD simulation. Numerical parameters used in this study are given in Table III.

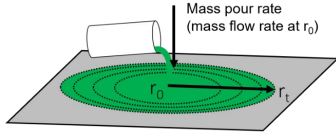


Fig. 7. Schematic illustration of the spill scenario [12].

Table I Pouring conditions for FLiNaK spreading simulation

Parameter	Value	Unit
Mass pour rate	20.2	g/s
Time of pouring	5.0	s
Initial temperature of FLiNaK	650	°C
Temperature of plate and atmosphere	20	°C
Plate length $\times$ width	0.15×0.15	m
Plate thickness	1.59×10^{-3}	m
Injection height above plate surface	0.01	m

Table II FLiNaK thermophysical properties used in the base case scenario

Parameter	Value	Unit
Density	2020	kg/m ³
Dynamic viscosity	3.18×10^{-3}	kg/(ms)
Specific heat	1952	J/(kg K)
Thermal conductivity	0.85	W/(m K)
Latent heat	1.661×10^6	J/kg
Surface tension coefficient	0.179	N/m
Contact angle	90	deg
Radiation emissivity	1.0	-

Table III Key numerical parameters used in the E-MPS simulations

Parameter	Value	Unit
Particle distance	5.0×10^{-4}	m
Time step	1.0×10^{-4}	s
Speed of sound	5	m/s
Ratio of effective radius to particle distance		
Laplacian	3.1	-
Gradient	3.1	-
Surface Tension	3.2	-

4. Results

As the molten salt spreads across the steel substrate, its temperature field evolves with time (Fig. 8). The highest temperature region remains near the point of impact, while the leading edge cools progressively as it comes into contact with the plate. FLiNaK has low thermal conductivity and high latent heat. Because of this, the solidification rate is low and does not impact the dynamic behavior of the salt. Five seconds after pouring ends, the salt is still mostly molten.

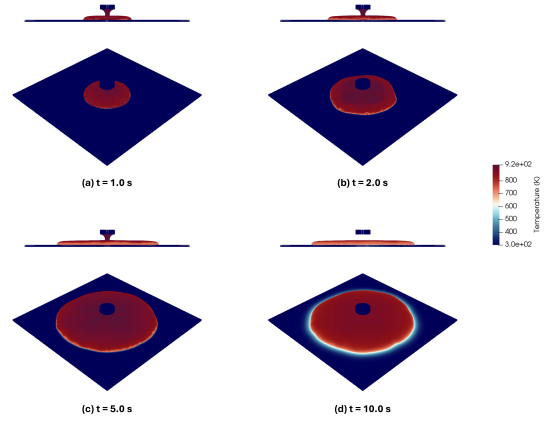


Fig. 8. Temperature field of the spreading molten salt on the substrate.

The E-MPS prediction consistently underestimates the leading edge radius: 6.1 cm versus 6.95 cm for MELTSPREAD (12.2 % lower) and 6.8 cm for Ansys Fluent (10.3 % lower) (Fig. 9).

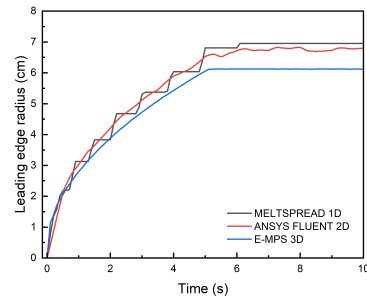


Fig. 9. Comparison of the molten salt leading edge radius between the E-MPS, MELTSPREAD and Ansys Fluent.

Heat transfer is dominated by conduction (Fig. 10). It rises sharply while hot salt is being poured onto the plate, then slows once pouring stops and the plate warms toward the salt temperature. Radiation grows steadily after pouring stops. By $t = 10$ s, the total heat transfer reaches about 1.46×10^4 J, split between conduction (1.04×10^4 J, 71 %) and radiation (4.2×10^3 J, 29 %).

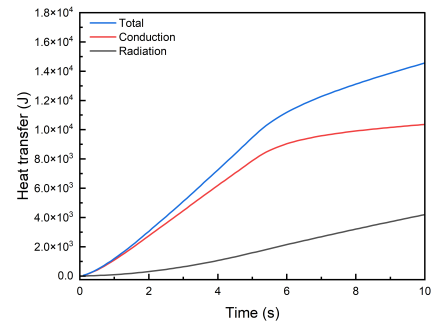


Fig. 10. The total, conductive and radiative heat transfer.

The GPU-accelerated implementation scales linearly with new particles being generated throughout the simulation (Fig. 11). Wall-clock time per step rises from 0.04 s at 390000 particles to 0.19 s at 770000 particles ($R^2 = 0.998$).

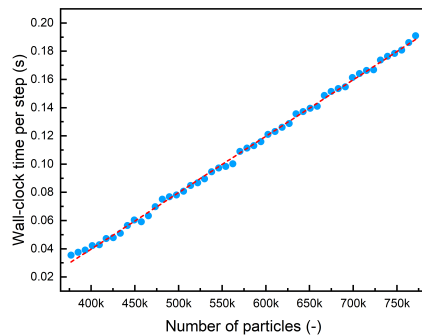


Fig. 11. Computational performance scaling.

5. Conclusions

The E-MPS method presented in this study, with phase change and surface tension models, gives stable results and captures the full 3D scope of the phenomenon. Linear scalability makes the framework capable of running larger simulations in a GPU environment within practical time.

However, the leading edge radius was underestimated compared to MELTSPREAD and Ansys Fluent. The reason is the surface tension potential-based method used in this study. It hinders the flow and neglects the effect of inertia. Inertia would otherwise lead to further spreading of the molten salt. Future research, including an updated surface tension model, is needed.

Acknowledgement

This research was supported by the 2026 Research Fund of the KEPKO International Nuclear Graduate School (KINGS), the Republic of Korea.

REFERENCES

- [1] IAEA, Status of Molten Salt Reactor Technology, Technical Reports Series No. 489, 2023.
- [2] D.E. Holcomb, G.F. Flanagan, R.J. Belles, and W.P. Poore III, A Safety and Licensing Roadmap to Identify the Research and Development Gaps of Commercial Molten Salt Reactors, Oak Ridge National Laboratory, 2018.
- [3] S.E. Beall, P.N. Haubenreich, R.B. Lindauer, and J.R. Tallackson, MSRE Design and Operations Report, Part V, Reactor Safety Analysis Report, Oak Ridge National Laboratory, 1964.
- [4] H. Gotoh and A. Khayyer, On the state-of-the-art of particle methods for coastal and ocean engineering, Coastal Engineering Journal, vol. 60, pp. 79–103, 2018.
- [5] S. Koshizuka and Y. Oka, Moving particle semi-implicit method for fragmentation of incompressible fluid, Nuclear

Science and Engineering, vol. 123, pp. 421–434, 1996.

- [6] A. Yamaji and X. Li, Development of MPS method for analyzing melt spreading behavior and MCCI in severe accidents, Journal of Physics: Conference Series, vol. 739, 012002, 2016.
- [7] Jubaidah, G. Duan, A. Yamaji, C. Journeau, L. Buffe, and J.F. Haquet, Investigation on corium spreading over ceramic and concrete substrates in VULCANO VE-U7 experiment with moving particle semi-implicit method, Annals of Nuclear Energy, vol. 141, 107266, 2020.
- [8] L. Zhao, M. Punetha, W. Ma, P. Isaksson, S. Bechta, S. Lomperski, M. Farmer, and J. Licht, Application of moving particle semi-implicit method on simulating melt spreading within OECD/ROSAU project, Nuclear Engineering and Design, vol. 427, 113447, 2024.
- [9] T. Kawahara and Y. Oka, Ex-vessel molten core solidification behavior by moving particle semi-implicit method, Journal of Nuclear Science and Technology, vol. 49, no. 12, pp. 1156–1164, 2012.
- [10] Y. Yamada, M. Sakai, S. Mizutani, S. Koshizuka, M. Oochi, and K. Murozono, Numerical simulation of three-dimensional free-surface flows with explicit moving particle simulation method, Transactions of the Atomic Energy Society of Japan, vol. 10, pp. 185–193, 2011.
- [11] A. Shakibaeinia and Y.-C. Jin, A weakly compressible MPS method for modeling of open-boundary free-surface flow, International Journal for Numerical Methods in Fluids, vol. 63, pp. 1208–1232, 2010.
- [12] S. Thomas, J. Jackson, and M. Farmer, Argonne National Laboratory, Modeling Molten Salt Spreading and Heat Transfer Using MELTSPREAD – An Uncertainty Analysis, 2021.
- [13] Y.H. Chun, J.H. Park, Y.S. Cho, S.M. An, S.I. Kim, J. Song, and S.J. Kim, Numerical assessment of NaCl-KCl-UCI3 spreading and solidification behavior in a salt spill accident, Nuclear Engineering and Technology, vol. 58, 104358, 2026.
- [14] S. Arrhenius, The viscosity of solutions, Biochemical Journal, vol. 11, pp. 112–133, 1917.
- [15] M. Kondo, S. Koshizuka, and M. Takimoto, A surface tension model using inter-particle potential force in Moving Particle Semi-implicit method, Transactions of The Japan Society for Computational Engineering and Science, vol. 2007, 20070021, 2007 (in Japanese).
- [16] S. Green, NVIDIA Corporation, Particle Simulation Using CUDA, Technical Report, 2010.