

Refined energy-conserving dissipative particle dynamics model with temperature-dependent properties and its application in solidification problem

K. C. Ng*

*National Center for Theoretical Sciences (NCTS), National Taiwan University, Taipei, Taiwan
and Department of Mechanical Engineering, Universiti Tenaga Nasional, Jalan IKRAM-UNITEN, 43000 Kajang, Selangor, Malaysia*

T. W. H. Sheu†

Center for Advanced Study on Theoretical Sciences (CASTS), National Taiwan University, Taipei, Taiwan

(Received 13 April 2017; revised manuscript received 17 July 2017; published 4 October 2017)

It has been observed previously that the physical behaviors of Schmidt number (Sc) and Prandtl number (Pr) of an energy-conserving dissipative particle dynamics (eDPD) fluid can be reproduced by the temperature-dependent weight function appearing in the dissipative force term. In this paper, we proposed a simple and systematic method to develop the temperature-dependent weight function in order to better reproduce the physical fluid properties. The method was then used to study a variety of phase-change problems involving solidification. The concept of the “mushy” eDPD particle was introduced in order to better capture the temperature profile in the vicinity of the solid-liquid interface, particularly for the case involving high thermal conductivity ratio. Meanwhile, a way to implement the constant temperature boundary condition at the wall was presented. The numerical solutions of one- and two-dimensional solidification problems were then compared with the analytical solutions and/or experimental results and the agreements were promising.

DOI: [10.1103/PhysRevE.96.043302](https://doi.org/10.1103/PhysRevE.96.043302)

I. INTRODUCTION

To date, various particle-based methods have been developed to solve complex fluid and/or solid mechanics problems. Some notable particle-based methods are smoothed particle hydrodynamics (SPH) [1–3], moving particle semi-implicit (MPS) [4–18], dissipative particle dynamics (DPD), etc. Dissipative particle dynamics (DPD) is a particle-based method originally proposed by Hoogerbrugge and Koelman [19]. In general, the particle considered in DPD represents a cluster of atoms or molecules (coarse grained), instead of an individual atom or molecule considered in molecular dynamics (MD). Therefore, DPD can be used to compute problems involving spatial and time scales which are larger than those considered in MD such as fiber suspension [20], microswimmer [21,22], blood flow [23], polymer dynamics [24,25], DNA suspensions [26,27], microfluidic systems [28], etc. A more thorough review on the applications of DPD can be found in the open literature [29–32]. In fact, the coarse-grained DPD equations can be derived from the microscopic system (see details in [33]). Espanol and Revenga [34], on the other hand, added thermal fluctuations on the macroscopic smoothed particle hydrodynamics (SPH) governing equations [i.e., the smoothed dissipative particle dynamics (SDPD) method]. These fluctuations are absent if a continuum problem is simulated.

In order to extend the numerical framework of DPD to handle nonisothermal flow, the energy-conserving dissipative particle dynamics (eDPD) method has been developed [35,36]. More recently, Chaudhri and Lukes [37] have extended the formulations of the eDPD method to compute multicomponent flow. To date, eDPD has been used in many fluid flow applications involving heat transfer such as the heat conduction

problem [38], heat convection problem [39], conjugate heat transfer problem [40], conduction problem in nanocomposite [41], thermophoretic microswimmer problem [21], etc. By combining the enthalpy method with eDPD, problems involving phase change (i.e., melting) have been addressed by Willemsen *et al.* [42,43] and more recently by Johansson *et al.* [44].

It is well known that the thermophysical property of a real fluid changes with respect to temperature. In fact, for common liquids such as water, its dynamic viscosity decreases as temperature increases. In order to capture this physical behavior using eDPD, Li *et al.* [45] recently devised a temperature-dependent formulation for the exponent s (denoted as “Li’s formulation” in this paper) appearing in the weight function of dissipative force. Basically, they have enlarged the cutoff radius (r_c) from its conventional value ($=1.0$) to $r_c = 1.58$. Coupled with their proposed temperature-dependent exponent, $s(T) = C_1 + C_2(T^2 - 1)$ where T is the dimensionless temperature, they have somehow reproduced the Schmidt number and the Prandtl number of liquid water at various temperatures. In their work, the values of C_1 and C_2 were prescribed as 0.41 and 1.9, respectively. The details on how to choose the values of r_c , C_1 , and C_2 , however, were not given. More recently, Abu-Nada [46] discussed the difficulty in extending Li’s formulation to other fluids. In order to resolve this issue, Abu-Nada [46] incorporated the temperature-dependent viscosity ratio in the w_D weight function of Fan *et al.* [27]. Although the extension of this method to handle other fluids is straightforward, it remains an open question whether this method is able to reproduce the Schmidt number of a real fluid at different temperatures. Yamada *et al.* [47] have recently highlighted the importance of modeling the freezing behavior in microfluidic devices such as phase-change valves and proton exchange membrane fuel cells (PEMFCs) in order to control the ice formation. Following this, Johansson *et al.* [44] adopted Li’s formulation in modeling the

*Corresponding author: ngkhaiching2000@yahoo.com

†twshsheu@ntu.edu.tw

freezing problem occurring in a flow channel, prior to studying the more complex flow in PEMFC. They used a larger r_c in their two-dimensional (2D) eDPD simulation ($r_c = 1.81$); however, the rationale behind this selection is unclear.

In the current paper, we propose an alternative temperature-dependent formulation of the exponent s . The selection of a proper cutoff radius will be outlined based on the kinetic theory, and the procedures in obtaining the s values at different temperatures will be described. Due to the fact that the physical property of liquid water below 0 °C (i.e., supercooled water) is available, we have further extended this method to cover a wider temperature range, i.e., $253.15 \text{ K} < T^R < 373.15 \text{ K}$, where T^R is the temperature with a physical unit (K). Our formulation was then used to simulate a variety of flow problems involving solidification. At this point, we will introduce the concept of ‘‘mushy’’ particle to model the eDPD particles in the transition zone. In order to properly represent the Dirichlet temperature boundary condition imposed at the wall, we have modified the original method of Willemsen *et al.* [42] by simply fixing the ghost particles in the wall, and their temperatures were interpolated from the neighboring fluid particles. Therefore, it is unnecessary to update the positions of ghost particles in our current implementation. We have further validated the eDPD method in the 2D solidification problem, where analytical and experimental solutions are available.

II. MATHEMATICAL MODEL

The governing equations of the energy-conserving dissipative particle dynamics (eDPD) model have been often reported in the literature. Here, these equations are rewritten for completeness. The equation of motion of an eDPD particle i can be expressed as

$$m_i \frac{d\vec{v}_i}{dt} = \vec{F}_i = \sum_{i \neq j} \vec{F}_{ij}^C + \vec{F}_{ij}^D + \vec{F}_{ij}^R + \vec{F}^E, \quad (1)$$

$$\frac{d\vec{r}_i}{dt} = \vec{v}_i, \quad (2)$$

where \vec{F}_{ij}^C is the conservative force vector, consisting of a soft repulsive force due to the interaction potential between particles. \vec{F}_{ij}^D is the dissipative force vector which dissipates the thermal kinetic energy of the system and reduces the velocity difference between the particles. On the other hand, the random force vector \vec{F}_{ij}^R generates a stochastic force on an eDPD particle. The dissipative and random forces are acting as a heat sink and a heat source, respectively. Their combination must satisfy the fluctuation-dissipation theorem in order to satisfy the energy balance within the fluid system. \vec{F}_{ij}^E is any external force acting on the particles, such as spring force, etc. In the current work, all particles have the same mass $m (=1.0)$, and the velocity vector \vec{v}_i can be evaluated by integrating Eqs. (1) and (2) with respect to time by using the velocity-Verlet scheme proposed by Groot and Warren [48]. Other numerical integrators can be found in [49]. The summation operator appearing in Eq. (1) was performed on all neighboring eDPD particles lying within the cutoff radius r_c . Mathematically, these force components are

expressed as

$$\vec{F}_{ij}^C = a_{ij} w_C(r_{ij}) \vec{e}_{ij}, \quad (3)$$

$$\vec{F}_{ij}^D = -\gamma_{ij} w_D(r_{ij}) (\vec{e}_{ij} \cdot \vec{v}_{ij}) \vec{e}_{ij}, \quad (4)$$

$$\vec{F}_{ij}^R = \sigma_{ij} w_R(r_{ij}) \xi_{ij} \Delta t^{-1/2} \vec{e}_{ij}. \quad (5)$$

Here, r_{ij} is the distance between particles i and j . \vec{e}_{ij} is the unit vector pointing from particle j to i . \vec{v}_{ij} is the velocity difference defined as $\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$. a_{ij} is the repulsive force parameter expressed as $a_{ij} = 75 k_B T_{ij} / \rho$ where k_B is the Boltzmann constant. T_{ij} is the effective temperature and ρ is the number density. Li *et al.* [45] have defined $T_{ij} = (T_i + T_j)/2$ in their work. However, we followed the approach of Fedosov *et al.* [21] by using a constant value of T_{ij} , i.e., $T_{ij} = (T_H + T_C)/2$, where T_H and T_C are hot and cold temperatures in the system, respectively. It is worth mentioning here that the constant parameter (i.e., 75) appearing in a_{ij} is obtained upon mapping with the compressibility of water. In order to simulate the interaction of different fluids, the Flory-Huggins parameter can be used [48,50]. Recently, Arai *et al.* [51] have increased the constant parameter appearing in the repulsive force parameter in order to model the hydrophobic nature of fluids in a vesicle system.

In order to satisfy the fluctuation-dissipation theorem, conditions such as $\sigma_{ij}^2 = 4\gamma_{ij} k_B T_i T_j / (T_i + T_j)$ and $w_D = w_R$ must be satisfied. Here, γ_{ij} and σ_{ij} are the intensities of dissipative and random forces, respectively. The weight functions such as $w_C = 1 - r/r_c$ and $w_D = (1 - r/r_c)^s$ (where $s = 2$) have been commonly adopted. ξ_{ij} is the symmetric random number with zero mean and unit variance, i.e., $\xi_{ij} = \xi_{ji}$. Qiao and He [41] have previously reported that the result simulated based on the uniform random number was similar to that of the Gaussian random number. Due to the fact that the generation of a uniform random number is computationally cheaper, the uniform random number was implemented in the current work.

In order to determine the temperature (T) of an eDPD particle, we have solved the energy equation as well:

$$\frac{de_i}{dt} = q_i = \sum_{i \neq j} q_{ij}^C + q_{ij}^V + q_{ij}^R + Q_{\text{ext}}, \quad (6)$$

where e_i is the internal energy of particle i , i.e., $e_i = C_v T_i$, and C_v is the specific heat. The terms such as collisional heat flux q_{ij}^C , viscous heat flux q_{ij}^V , and random heat flux can be written as

$$q_{ij}^C = \sum_{i \neq j} k_{ij} w_{CT} (T_i^{-1} - T_j^{-1}), \quad (7)$$

$$q_{ij}^V = \frac{1}{2C_v} \sum_{i \neq j} \{ w_D [\gamma_{ij} (\vec{e}_{ij} \cdot \vec{v}_{ij})^2 - \sigma_{ij}^2 / m_i] - \sigma_{ij} w_R (\vec{e}_{ij} \cdot \vec{v}_{ij}) \xi_{ij} \Delta t^{-1/2} \}, \quad (8)$$

$$q_{ij}^R = \sum_{i \neq j} \beta_{ij} w_{RT} \Delta t^{-1/2} \xi_{ij}^e. \quad (9)$$

Any external heat source can be included in Q_{ext} . The term k_{ij} indicates the strength of the collisional heat flux, which is related to the mesoscopic heat friction k_{ij} as

$k_{ij} = C_v^2 \kappa_{ij} (T_i + T_j)^2 / 4k_B$. As reported by Zhang *et al.* [40], the mesoscopic heat friction k_{ij} is treated as heat conductivity in the mesoscopic scale; therefore, the harmonic mean procedure can be used to express the equivalent heat friction between two particles: $\kappa_{ij} = 2\kappa_i \kappa_j / (\kappa_i + \kappa_j)$. Inspired from the mapping of macroscopic thermal conductivity with eDPD parameters [52], Li *et al.* [45] have expressed the mesoscopic heat friction in terms of the Prandtl number which is experimentally available (see Eq. (15) in [45]). Meanwhile, the strength of the random heat flux β_{ij} is related to k_{ij} as $\beta_{ij} = \sqrt{2k_B k_{ij}}$. The weight functions of collisional and random heat fluxes, $w_{CT} = (1 - r/r_c)^2$ and $w_{RT} = (1 - r/r_c)$, were adopted. ξ_{ij}^e is the antisymmetric uniform random number with zero mean and unit variance, i.e., $\xi_{ij}^e = -\xi_{ji}^e$.

It is important to note that the variables defined above were written in their dimensionless forms. In the current work, the reference temperature T_*^R was 300 K. Here, the superscript R signifies the quantity with physical units and the subscript $*$ indicates the reference quantity. We have set $\rho = 4$ and $C_v = 100\,000$ in all simulations. The scaling procedure that was used to produce the dimensionless variables of water can be found in [45]. In general, the eDPD model is more coarse grained when C_v increases [41].

III. THERMOPHYSICAL PROPERTIES OF WATER

The exponent s appearing in w_D is normally prescribed as 2.0 in DPD. This is permissible if the momentum diffusion is of the same order as the particle diffusion (or self-diffusion) in a real fluid, i.e., the Schmidt number (Sc) is ~ 1.0 [48]. However, for a real fluid such as water, $Sc \sim 350$ at $T_*^R = 300$ K, indicating that the simulated Sc is lower than the actual Sc by several orders of magnitude if the standard DPD parameters are employed. This condition occurs due to several reasons. Apart from the soft interaction between the DPD particles [53], shearing dissipation may disappear even though two DPD particles are close to each other [27]. In general, the fluid is behaving like a gas if Sc is low, or like a liquid if Sc is large [54]. Ripoll *et al.* [55] observed that the correct hydrodynamic behavior of suspended particles can be recovered only at large Sc (collective regime). Symeonidis *et al.* [56] compared the DPD results with the experimental data and concluded that the underprediction of Sc would degrade the accuracy of the DPD simulation. Fan *et al.* [27] found that the conventional DPD method was unable to simulate the flow of a real fluid in a complex channel because its dynamic response is too slow (low Sc).

From the linearized Fokker-Planck equation (see [57]), the hydrodynamic variables such as kinematic viscosity ν and self-diffusivity D can be written as follows:

$$\nu = \frac{k_B T}{2} \left(\frac{w_0 t_w^2}{d+2} + \frac{1}{w_0} \right), \quad (10)$$

$$D = \frac{k_B T}{m w_0}. \quad (11)$$

Here, $w_0 = \gamma \rho [w_D]_r / m d$ and $t_w^2 = m [r^2 w_D]_r / k_B T [w_D]_r$. Depending on the flow dimensionality d , the bracketed term, say $[\phi]_r$, is indeed an integral operator: $[\phi]_r = \int_0^{r_c} \phi 2\pi r dr$

for $d = 2$ or $[\phi]_r = \int_0^{r_c} \phi 4\pi r^2 dr$ for $d = 3$. The upper bound of the integrand is the cutoff radius, r_c .

By employing $w_D = (1 - r/r_c)^s$ and integrating terms such as $[w_D]_r$ and $[r^2 w_D]_r$, the kinematic viscosity (ν) and self-diffusivity (D) can now be determined as a function of s . Here, we focus on a two-dimensional problem ($d = 2$):

$$\nu = \frac{3\pi\gamma\rho r_c^4}{4(s+1)(s+2)(s+3)(s+4)} + \frac{k_B T (s+1)(s+2)}{2\pi\gamma\rho r_c^2}, \quad (12)$$

$$D = \frac{k_B T (s+1)(s+2)}{\pi\gamma\rho r_c^2}. \quad (13)$$

Thus the Schmidt number can be written as follows:

$$Sc = \frac{\nu}{D} = \frac{1}{2} + \frac{3\pi^2\gamma^2\rho^2 r_c^6}{4k_B T (s+1)^2 (s+2)^2 (s+3)(s+4)}. \quad (14)$$

From Eq. (14), it is now straightforward to devise appropriate methods to increase the Schmidt number of the DPD fluid. One may decrease the temperature T as done in [54]. Alternatively, Fan *et al.* [27] have used a smaller value of s , i.e., $s = 0.5$. Methods such as increasing r_c or ρ can be adopted as well at the expense of more computational cost. Finally, the idea of increasing the dissipative force intensity γ can be pursued; however, the random force intensity σ must be increased (hence reduced Δt) to satisfy the fluctuation-dissipation theorem.

For nonisothermal flow, the method of reducing s seems to be the most practical approach, as the associated increase of computational cost is not significant at all. For example, Li *et al.* [45] have devised a special temperature-dependent function of s , i.e., $s(T) = 0.41 + 1.9(T^2 - 1)$, to reproduce the Schmidt number of water for temperature ranging from $273 < T^R < 373$ K. Meanwhile, they have increased the cutoff radius r_c to 1.58. The rationales behind the development of their function $s(T)$ and the choice of r_c are, however, unclear. This would unfortunately complicate the process of devising the proper $s(T)$ function for other working fluids. More recently, Abu-Nada [46] proposed a new weight function for dissipative force by simply multiplying the temperature-dependent viscosity ratio of a fluid with the weight function suggested by Fan *et al.* [27], i.e., $w_D = \frac{\mu(T)}{\mu_c} (1 - r/r_c)^{1/2}$, where μ_c is the reference dynamic viscosity. However, the ability of this new model to reproduce the Schmidt number of a real fluid remains unknown. By relying on the same exponent $s(T)$ developed by Li *et al.* [45], Johansson *et al.* [44] have computed the 2D mesoscopic solidification problem using eDPD. They enlarged the cutoff radius r_c to 1.81 while keeping the standard DPD parameters unchanged, i.e., $\gamma = 4.5$ and $\rho = 4$. The simulated Schmidt numbers of Li *et al.* [45] and Johansson *et al.* [44] at different temperatures are shown in Fig. 1. As seen, the computed Schmidt numbers based on the $s(T)$ function of Li *et al.* [45] vary considerably with those measured experimentally, especially at the low-temperature regime. On the other hand, the overprediction of the Schmidt numbers simulated by Johansson *et al.* [44] is quite discernible in the medium-temperature range: $291 < T^R < 312$ K (or $0.97 < T < 1.04$).

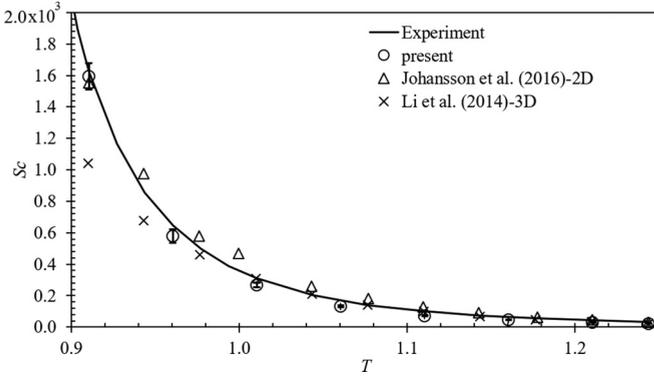


FIG. 1. Comparison of Schmidt numbers of water for $0.91 < T < 1.24$. For the present eDPD simulations, the error bars are obtained by repeating the simulations five times. $r_c = 1.96$, $\gamma = 8.0$, $\rho = 4$.

Now the main question is how to choose the DPD parameters in a more systematic manner in order to reproduce the Schmidt number of water. Firstly, it is important to note that the cutoff radius r_c is a constant parameter (fixed during the eDPD computation). Therefore, Eq. (14) can be rearranged in a way leading to

$$r_c = \left[\frac{(\text{Sc} - \frac{1}{2})4k_B T (s+1)^2 (s+2)^2 (s+3)(s+4)}{3\pi^2 \gamma^2 \rho^2} \right]^{1/6}. \quad (15)$$

The plots of r_c against s for different temperatures are shown in Fig. 2. Here, we replaced the term Sc by the target value (i.e., experimental value of Schmidt number at a specific temperature T), and utilized the standard DPD

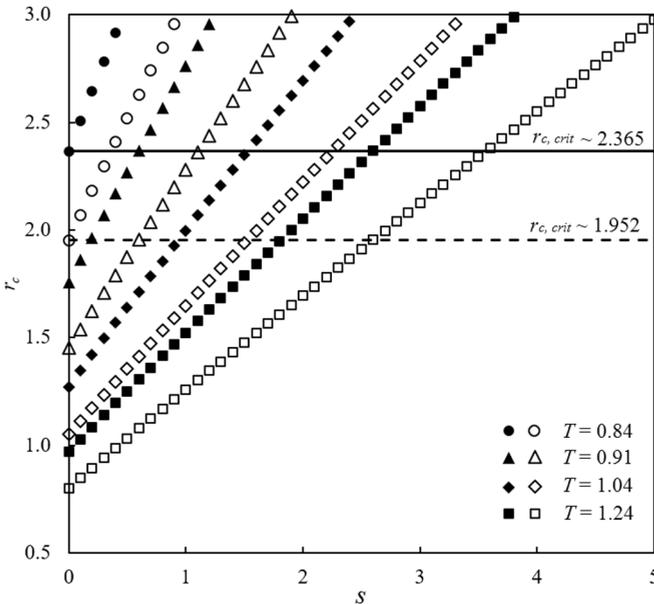


FIG. 2. The relationship between r_c and s at different temperatures. Solid polygons: $\gamma = 4.5$ and $\rho = 4$ ($r_{c,\text{crit}} \sim 2.365$). Empty polygons: $\gamma = 8.0$ and $\rho = 4$ ($r_{c,\text{crit}} \sim 1.952$). r_c must be chosen in such a way that $r_c > r_{c,\text{crit}}$.

parameters such as $\gamma = 4.5$ and $\rho = 4$ to generate the plots (represented by solid polygons) at four temperatures, e.g., $T = 0.84, 0.91, 1.04$, and 1.24 . Due to the fact that the thermophysical properties of supercooled liquid water are experimentally available (see [58–60]), we have considered a working temperature range: $253.15 < T^R < 373.15$ K (or $0.84 < T < 1.24$), which is larger than that considered by Li *et al.* [45].

As observed graphically from Fig. 2, once the r_c value is prescribed, the exponent s at a corresponding temperature T can be determined. It is found that exponent s is increasing with respect to temperature, which is agreeable to the original proposal of Li *et al.* [45]. Obviously, a low value of r_c is preferable in order to shorten the simulation time (the cost of force calculation is proportional to r_c^3 ; see [27]). As observed from Fig. 2, in order to ensure $s > 0$ (i.e., w_D is decaying within r_c) for the entire temperature range, the condition of $r_c > r_{c,\text{crit}}$ must hold. By using the standard DPD parameters (see plots consisting of solid polygons), it seems that $r_{c,\text{crit}} \sim 2.365$. Alternatively, the term $r_{c,\text{crit}}$ can be calculated from Eq. (15) by setting $s = 0$ and $\text{Sc} = 10356$ (i.e., the experimental value of Schmidt number at the lowest temperature within the range considered: $T = 0.84$).

Now, if one relies on the standard parameters, e.g., $\gamma = 4.5$ and $\rho = 4$, one may choose $r_c = 2.37$ (i.e., a value which is slightly larger than $r_{c,\text{crit}} \sim 2.365$) for the eDPD computation. However, this implies that particle information in at least three (the next integer of $r_{c,\text{crit}}$) neighboring cells of the local cell containing particle i must be processed during the interactive force/heat computation. Meanwhile, we have found that by increasing the dissipative force intensity γ to 8.0 and retaining $\rho = 4$, $r_{c,\text{crit}}$ can be reduced to 1.952 according to Fig. 2 (plots consisting of empty polygons) or Eq. (15). Correspondingly, due to the increase of γ (hence increase of σ), the time step size was reduced from the recommended value $\Delta t = 0.01$ (used in [44]) to $\Delta t = 0.007$ to ensure numerical stability. Therefore, the eDPD parameters, i.e., $r_c = 1.96$, $\gamma = 8.0$, $\rho = 4$, and $\Delta t = 0.007$, will be used in the test cases outlined in the current work unless stated otherwise.

Based on the selected eDPD parameters, it is now possible to determine the temperature-dependent function of s graphically for different values of temperature T (e.g., Fig. 2) or based on Eq. (15). We have opted to solve Eq. (15) numerically for s . Given the set of experimental values of Schmidt number Sc at a particular temperature T (34 of them), the exponents s were solved iteratively by using the eDPD parameters suggested above. We found that the quadratic function $s(T) = -6.3039T^2 + 19.594T - 12.032$ can be used to fit the discrete $s(T)$ data with the coefficient of determination $R^2 = 0.9997$.

In order to verify our model, we have computed the Schmidt number of water from eDPD simulations. Firstly, the self-diffusivity of eDPD fluid was measured by using the mean square displacement χ . The self-diffusivity D was then calculated from χ via $D = \lim_{t \rightarrow \infty} \chi/4t$. In order to compute the kinematic viscosity (ν) of an eDPD fluid, the periodic Poiseuille flow (PPF) method of Backer *et al.* [61] was employed. Following the concept of the PPF method, the periodic heat conduction (PHC) method was developed

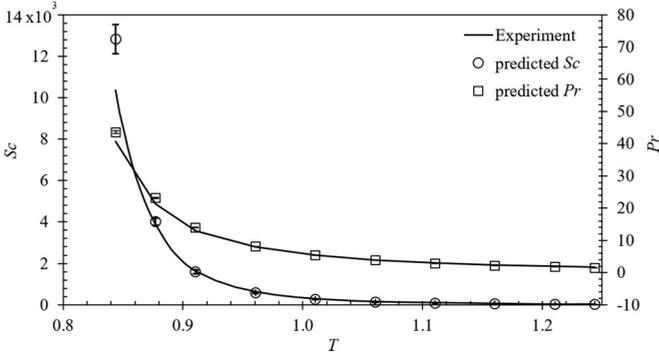


FIG. 3. Comparison of Schmidt and Prandtl numbers of water for $0.84 < T < 1.24$. The error bars of the present eDPD simulations are obtained by repeating the simulation five times. $r_c = 1.96$, $\gamma = 8.0$, $\rho = 4$.

by Li *et al.* [45] to determine the thermal diffusivity α of an eDPD fluid. The Prandtl number of an eDPD fluid can then be determined via $\text{Pr} = \nu/\alpha$.

A. Measured Schmidt and Prandtl numbers of eDPD fluid

Figure 1 shows the Sc values obtained by using the current approach. The values obtained by Li *et al.* [45] and Johansson *et al.* [44] were overlaid on the same figure as well for comparison purposes. As seen, among the three approaches, our predictions come closer to the measured values in general, especially for those in the low-temperature regime ($0.91 < T < 1.00$). We have repeated the numerical measurement of Sc (at each temperature) five times and found that the standard deviation of Sc was fluctuating between 3% and 7% of the averaged Sc value. On the other hand, at $T = 0.84$ (see Fig. 3), it is noticed that the difference between the predicted and the experimental values of Sc is quite discernible. At $T = 0.84$, our estimated s value is ~ 0.0068 . This somewhat small value of s implies that the changes of dissipative and random forces [functions of $w_D = (1-r/r_c)^s$] are more abrupt at $r = r_c$. This prompted Fan *et al.* [27] to choose a higher value of s (i.e., $s = 0.5$) in their flow computation. Therefore, if one intends to better reproduce the Schmidt number of supercooled water at $T = 0.84$, one might increase the values of r_c , γ , or ρ in order to get rid of the small value of s at $T = 0.84$ (supercooled water) at the expense of higher computational cost. This strategy, however, was not implemented in the current work.

Figure 3 compares the simulated Pr with the experimental data as well. In general, the agreement is quite promising except at $T = 0.84$ which may be attributed to the reason stated above ($s \sim 0$).

IV. SOLIDIFICATION PROBLEM

A. One-dimensional solidification in a semi-infinite plate

Having determined the eDPD parameters from Sec. III, i.e., $r_c = 1.96$, $\gamma = 8.0$, $\rho = 4$, and $\Delta t = 0.007$, we investigated the one-dimensional (1D) unsteady solidification problem in a semi-infinite plate which was previously studied by Johansson *et al.* [44] using eDPD. Following their geometric setup, a 2D rectangular domain of size 12×100 was built. The

initial temperatures of the eDPD particles were prescribed as $T_o = 0.95$. At $t = 0$, the temperature of the bottom wall was reduced to $T_b = 0.85$ and the liquid particles were allowed to freeze thereafter. Following the approach of Willemssen *et al.* [43], the equations of motion were solved only on nonsolid particles (i.e., eDPD particles were frozen once they became solid). The time evolution of the solid-liquid interface was then examined. In order to mimic the semi-infinite behavior of the problem, the height of the channel was set to a relatively large value in order to minimize the wall effect from the top boundary ($T_t = T_o = 0.95$). Periodic boundary conditions were prescribed at the left and right boundaries.

An analytical solution for this problem is available (see [62]), whereby the temperature at the solid and liquid regions can be expressed as follows:

$$T_S(y) = T_b + \frac{T_f - T_b}{\text{erf}(\lambda)} \text{erf}\left(\frac{y}{2\sqrt{\alpha_S t}}\right), \quad (16)$$

$$T_L(y) = T_t - \frac{T_t - T_f}{\text{erfc}(\lambda\sqrt{\alpha_{SL}})} \text{erfc}\left(\frac{y}{2\sqrt{\alpha_L t}}\right). \quad (17)$$

Here, the subscripts S and L denote solid and liquid, respectively. T_f is the freezing temperature (i.e., $T_f = 0.9105$). The term $(\cdot)_{SL}$ denotes the ratio $(\cdot)_S/(\cdot)_L$. In general, the constant λ determines the speed of the solid-liquid interface Ψ :

$$\Psi = 2\lambda\sqrt{\alpha_S t}. \quad (18)$$

As shown by Lunardini [62], the parameter λ can be solved numerically (iterative method) from the following implicit equation:

$$\frac{e^{-\lambda^2}}{\text{erf}(\lambda)} - \frac{k_{LS}\sqrt{\alpha_{SL}}(T_t - T_f)e^{-\alpha_{SL}\lambda^2}}{(T_f - T_b)\text{erfc}(\lambda\sqrt{\alpha_{SL}})} = \frac{L\lambda\sqrt{\pi}}{C_v(T_f - T_b)}, \quad (19)$$

where k is the thermal conductivity ($k = \rho C_v \alpha$). Note that the energy scale in eDPD is $k_B^R T_*^R$ (J). By considering the fact that the latent heat of water is $L^R = 334\,000$ J/kg, the dimensionless latent heat L can be computed by performing the scaling operation as follows:

$$L = \frac{L^R}{k_B^R T_*^R / m_*^R} = \frac{\rho^R (I_*^R)^3 L^R}{\rho k_B^R T_*^R}. \quad (20)$$

By substituting the latent heat of water ($L^R = 334\,000$ J/kg), the density of water $\rho^R = 1000$ kg/m³, the freezing temperature $T_f^R = 273.15$ K, and the Boltzmann constant $k_B^R = 1.3806 \times 10^{-23}$ J/K, the dimensionless latent heat can be computed from Eq. (20) as $L = 2.68 \times 10^4$. This value is similar to that obtained by Johansson *et al.* [44].

The simulation was executed until $t = 280$. In order to compare with the analytical solution, we fixed the values of α_S and α_L in the solid and liquid regions, respectively. Here, the thermal diffusivity in the liquid region α_L was prescribed as 1.1. The mesoscopic heat friction can then be calculated accordingly (see Eq. (15) in [44]). Besides that, we intend to examine the speed of the solid-liquid interface for different thermal diffusivity ratios as well, i.e., $\alpha_{SL} = 1.0, 2.0, 3.0$, and 4.08. Similar to most of the eDPD models, the parameters such as ρ and C_v were assumed to be temperature independent in the current work. Therefore, the case of $\alpha_{SL} = 4.08$ is analogous to the case where $k_{T=T_b(\text{ice})}^R / k_{T=T_t(\text{water})}^R = 4.08$.

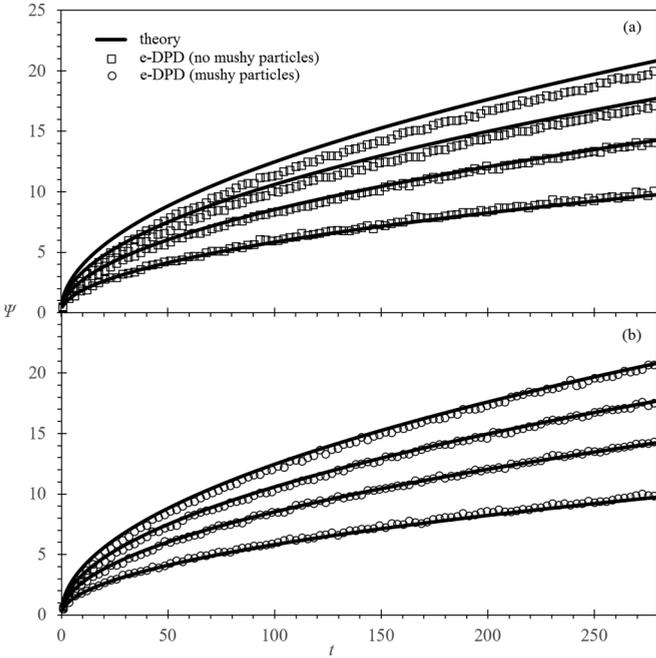


FIG. 4. Time evolutions of solid-liquid interface Ψ for cases (a) without mushy particle and (b) with mushy particles at different thermal diffusivity ratios (α_{SL}), i.e., 1.0, 2.0, 3.0, and 4.08. As α_{SL} increases, the speed $d\Psi/dt$ increases.

In order to study the solidification problem, we made a slight modification on the state equation initially proposed by Willemsen *et al.* [42] in their enthalpy method (via eDPD) for the melting problem. Here, the temperature of particle i was updated as follows:

$$T_i = \begin{cases} (e_i + L)/C_v & e_i < C_v T_f - L \\ T_f & C_v T_f - L \leq e_i \leq C_v T_f \\ e_i/C_v & e_i > C_v T_f \end{cases} \quad (21)$$

Generally, the state equation was used to describe the particle temperature in three distinct regions, i.e., solid region: $e_i < C_v T_f - L$; transition region: $C_v T_f - L \leq e_i \leq C_v T_f$; and liquid region: $e_i > C_v T_f$. According to the method outlined very recently by Johansson *et al.* [44], the states of the particles within the transition region remained, i.e., liquid or solid particles were retained in their original states when transition occurred.

We have attempted the approach suggested by Johansson *et al.* [44] and the results are shown in Fig. 4(a). In general, the speed of the solid-liquid interface increases with respect to the thermal diffusivity ratio α_{SL} . Here, the displacement of the solid-liquid interface Ψ was determined by identifying the locations of bins with temperature $T_{bin} = T_f$. The mean positions of these bins were then treated as the displacement of Ψ . Each cell was divided into five bins in the y direction, as shown in Fig. 5. As observed from Fig. 4(a), when $\alpha_{SL} = 1.0$ and 2.0, the eDPD results are quite close to the analytical solutions. Johansson *et al.* [44] argued that freezing took a longer time to start in their eDPD simulations due to the fact that a substantial amount of latent heat $L(L = 2.68 \times 10^4)$ must be subtracted before freezing takes place. However, such behavior was not observed in our current computation, and

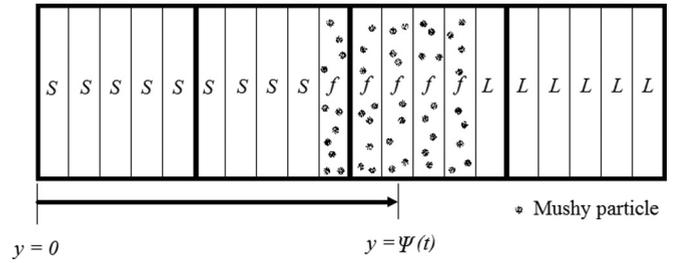


FIG. 5. The displacement measurement of solid-liquid interface $\Psi(t)$. S , L , and f denote the solid phase, liquid phase, and transition phase (mushy zone), respectively. Each cell (bounded by thick solid line) is divided into five bins in the y direction. The temperature in the transition zone is T_f .

freezing started almost instantaneously as suggested by the theoretical solutions. Meanwhile, it is noticed from Fig. 4(a) that as $\alpha_{SL} > 2.0$, the predicted speed of the solid-liquid interface differs from the theoretical solution, and the variation becomes more apparent as α_{SL} increases.

In fact, it has been well appreciated that the enthalpy method involves no explicit formulation of the solid-liquid interface. The front location is simply recovered based on the enthalpy (state equation). From the state equation, a particle is in full solid state when $e_i < C_v T_f - L$ and in full liquid state when $e_i > C_v T_f$. However, in the transition zone, a particle is partially solid (“mushy”). Figure 6 shows the thickness of the mushy layer (total size of bins with $T = T_f$) as time progresses for the case $\alpha_{SL} = 1.0$. As observed, the thickness increases to ~ 3 at the end of the simulation, in which its size is of the same order as the size of the cutoff radius r_c . Owing to this, we argued that the result could be improved if there is a more proper definition that can be used to describe the states of the particles within the transition zone. Based on the state equation, it is intuitive for us to define the solid fraction of

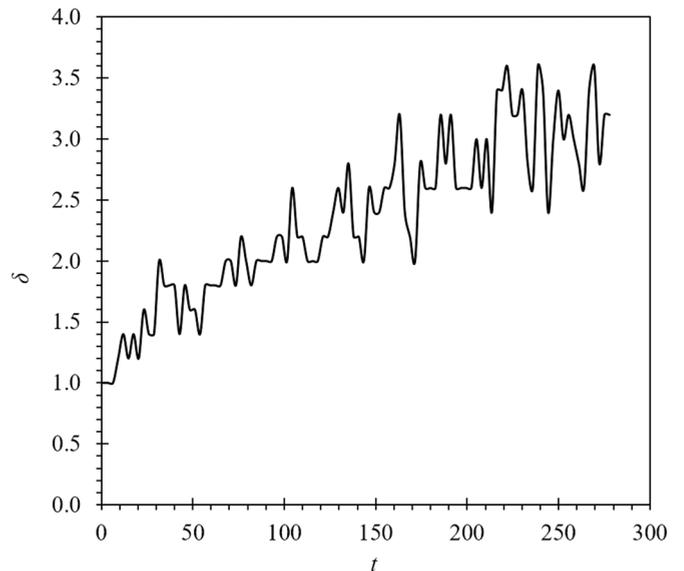


FIG. 6. The growth of mushy layer thickness as time elapses. $\alpha_{SL} = 1.0$.

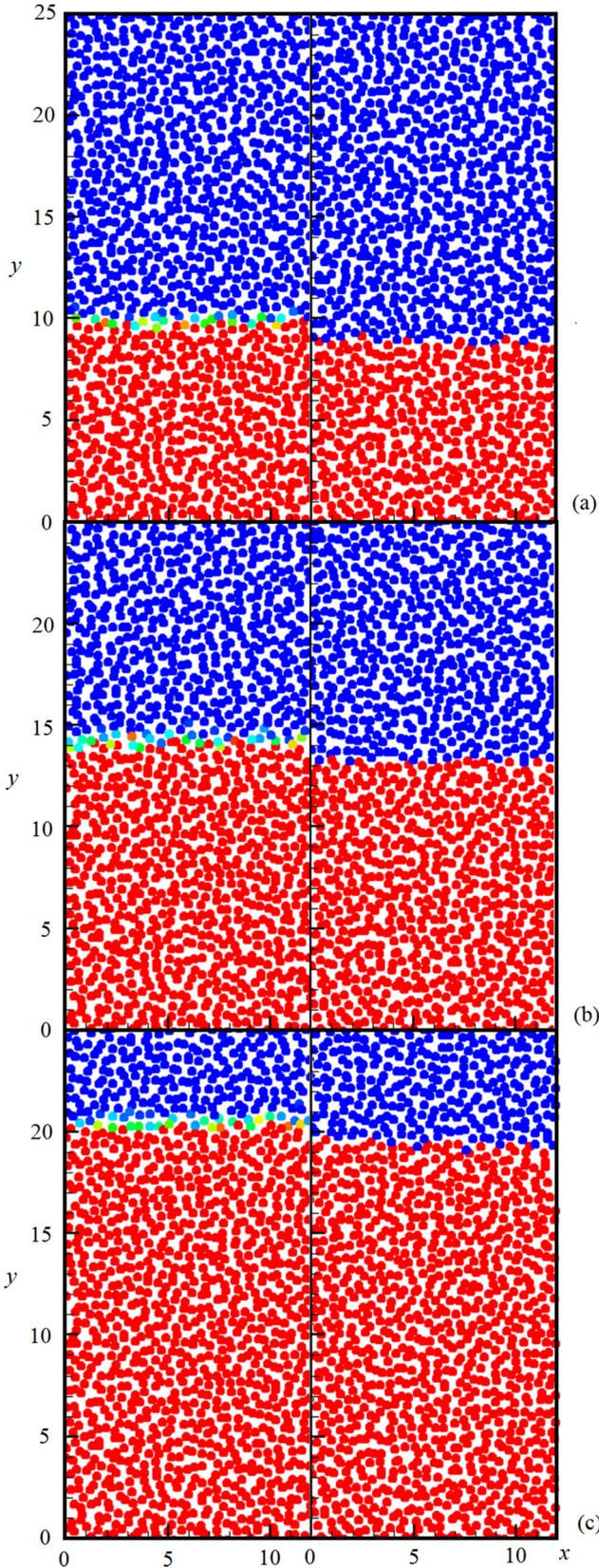


FIG. 7. Instantaneous positions of the eDPD particles at (a) $t = 70$, (b) $t = 140$, and (c) $t = 280$ colored by solid fraction φ_i . Red: solid phase ($\varphi_i = 1.0$). Blue: liquid phase ($\varphi_i = 0.0$). eDPD solutions with and without mushy particles are displayed on the left and right columns, respectively. $\alpha_{SL} = 4.08$.

particle i , namely, φ_i in the transition zone as follows:

$$\varphi_i = \frac{C_v T_f - e_i}{L}, \quad (22)$$

where φ_i is 0.0 and 1.0 for liquid and solid zones, respectively. In other words, in the transition or mushy zone, $0.0 < \varphi_i < 1.0$.

Now the underlying challenge is how to determine the thermal diffusivity of a mushy particle. According to Alexiadou and Solomon [63], the effective thermal diffusivity of a mushy particle relies heavily on the structure of the solid-liquid interface. Some formulations have been proposed in the numerical framework of finite-difference schemes. However, in the context of a particle method such as eDPD, it is unclear on how to define such a structure. Accordingly, we have linearly interpolated the thermal diffusivity of a mushy particle from α_S and α_L based on the computed solid fraction:

$$\alpha_i = \varphi_i \alpha_S + (1 - \varphi_i) \alpha_L. \quad (23)$$

A similar idea has been pursued in other particle methods such as the finite volume particle (FVP) method [64]. Figure 4(b) compares the positions of the solid-liquid interface for different thermal diffusivity ratios. As seen, the agreements between the eDPD results and the theoretical solutions are promising. The instantaneous positions of the eDPD particles are shown in Fig. 7 for different time levels. As observed, the predicted speed of the solid-liquid interface is lower if no mushy particle is considered in the flow computation.

We have further compared our computed temperatures at $t = 140$ with the theoretical solutions. Despite some statistical fluctuations, the predictions show good agreement with the theoretical solutions in general as shown in Fig. 8. We noticed that the near-wall temperature profile reported by Johansson *et al.* [44] was not reproduced correctly as their predicted temperature experienced a sharp increase near the wall (cf. Fig. 3 in [44]). Willemsen *et al.* [42] previously addressed this problem by reflecting the near-wall fluid particles about the wall axis. The new reflected particle was then assigned a temperature

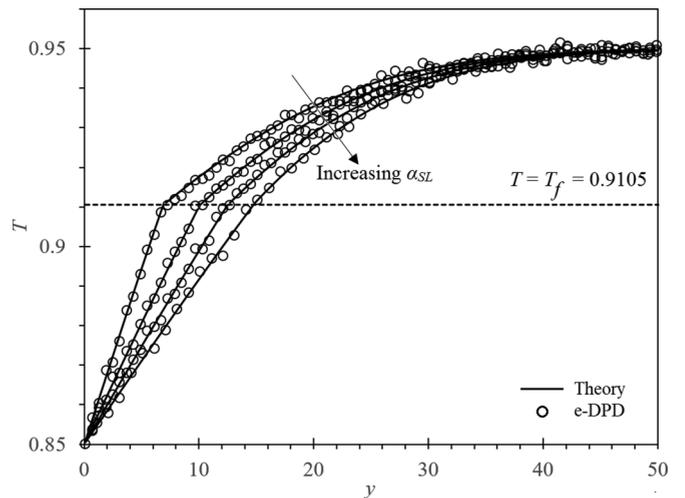


FIG. 8. Temperature distributions along the y direction for $\alpha_{SL} = 1.0, 2.0, 3.0,$ and 4.08 . eDPD method with mushy particle is used. Results at $t = 140$ are shown.

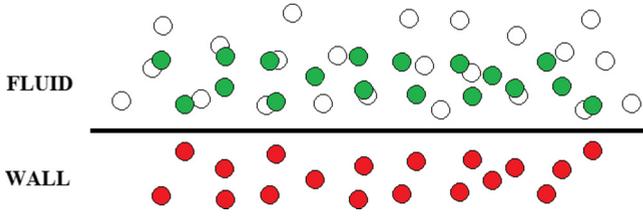


FIG. 9. Wall boundary condition. Red: wall particles. Green: Imaginary particles (reflection from wall particles about the wall boundary). Empty circle: interior fluid particle. Temperatures of imaginary particles are interpolated from interior fluid particles.

value in which the mean temperature of the original and the new particles equals the prescribed wall temperature. This strategy works well; however, the mirror particles must be generated at every time step due to the random motion of interior fluid particles. In the current work, we generated an extra layer of wall particles located within the wall region and their positions were fixed during the computation. In order to obtain their temperatures, we reflected the wall particles about the wall axis to form a new internal “imaginary” particle. Note that the positions of these imaginary particles were nonvarying because the positions of the wall particles were fixed. The temperature value of the wall particle was then assigned accordingly similar to that proposed by Willemsen *et al.* [42] by assuring that the mean temperature of the wall particle and the imaginary particle equals the prescribed wall temperature. Here, the temperature of the imaginary particle was interpolated from the interior eDPD particles by using the moving least square (MLS) method [65]. The schematic diagram of our implementation of the boundary condition is shown in Fig. 9. As reported in Fig. 8, the predicted near-wall temperature is reproduced correctly for different α_{SL} , which is consistent with the boundary condition ($T_b = 0.85$) imposed.

Finally, in order to eliminate the statistical noises, we have performed averaging on the unsteady temperature profiles for the case $\alpha_{SL} = 4.08$ based on 15 simulations generated from different random seeds. The results are shown in Fig. 10. As seen, there is a noticeable perturbation of temperature profile near the interface location due to the abrupt change of thermal diffusivity in the transition zone as practiced in the original method of Johansson *et al.* [44]. Meanwhile, it is appealing to note that the numerical results obtained by using a mushy particle come closer to the analytical solution.

B. 2D solidification problem

Next, we simulated the 2D solidification problem in a square cavity.

Rathjen and Jiji [66] have solved the 2D solidification problem on a semi-infinite domain analytically. This problem is governed by the Stefan number and the dimensionless initial temperature T_o^* expressed as follows:

$$St = \frac{C_v(T_f - T_w)}{L}, \tag{24}$$

and

$$T_o^* = \frac{T_o - T_f}{T_f - T_w}, \tag{25}$$

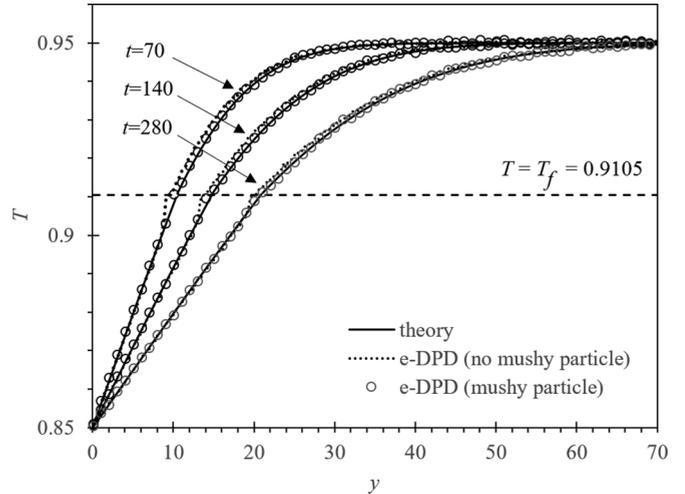


FIG. 10. Temperature distributions along the y direction for $\alpha_{SL} = 4.08$ at different time levels. eDPD methods with and without mushy particles are used. The temperatures are averaged by 15 sample data.

respectively. Here, the thermal diffusivity of $\alpha = 1.1$ was applied for both solid and liquid particles, and we have set $St = 0.222$ and $T_o^* = 2.0$. By employing the latent heat computed earlier for water, i.e., $L = 26800$, and the corresponding freezing temperature, $T_f = 0.9105$, the wall and initial temperatures can be calculated accordingly as $T_w = 0.8509$ and $T_o = 1.0296$, respectively.

While $t > 0$, the wall temperature was dropped to T_w (below the freezing temperature) to initiate the solidification process. Figure 11 shows the solidification process in the square domain. As seen, the solid-liquid interface shifts inward to the center of the square domain. The predicted dimensionless positions, i.e., $x^* = x/\sqrt{4\alpha_S t}$ and $y^* = y/\sqrt{4\alpha_S t}$ of the mushy particles for region $(0 < x < 15, 0 < y < 15)$ at different time levels, are compared to the analytical solution shown in Fig. 12. It is interesting to note that the numerical data collapse on the analytical solution for $t < 21$, confirming the similarity nature of the solidification problem considered here. The dimensionless positions of the mushy particles for $t > 21$ are shown in Fig. 13. Now, as expected, the simulated data no longer follow the semi-infinite theoretical solution due to the Dirichlet boundary conditions imposed at the top and right walls. As seen from Fig. 11(a), the shape of the solid-liquid interface mimics a square with small rounded edges at the corners initially. As time progresses, these rounded edges develop and the interface comes closer to a circle as witnessed in Fig. 11(d). In fact, the interface is thickening as seen in Figs. 13(a) and 13(b) as it is moving towards the center. As time progresses, mushy particles start to form even in the vicinity of the center of the square domain as seen in Fig. 13(c), forming a somewhat “circularlike” transition zone. Finally, this circularlike transition region shrinks as seen from Figs. 13(c)–13(f), signifying the end of the solidification process. Due to the symmetry of the domain shape, the dimensionless positions of the mushy particles are symmetric about the line $y^* = x^*$ illustrated in Figs. 13(a)–13(f). The change of the number of mushy particles is reported in Fig. 14.

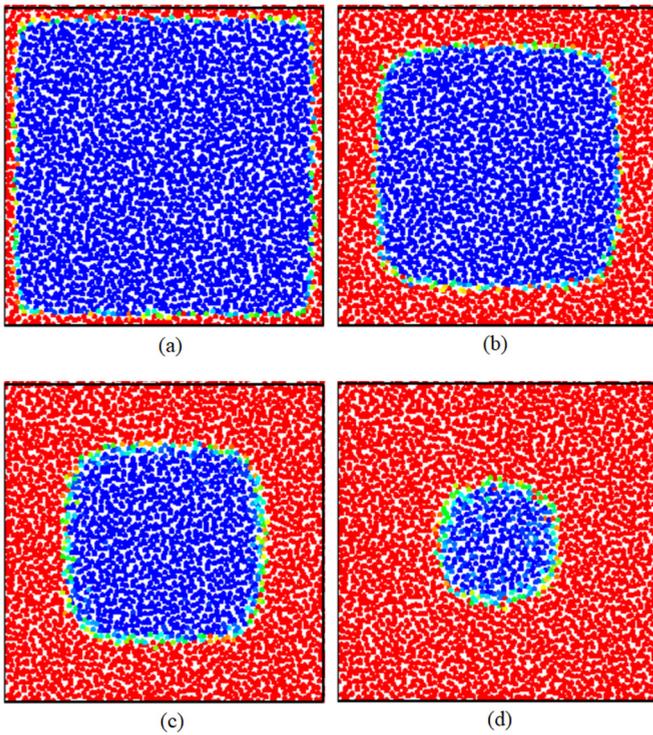


FIG. 11. The solidification process in a 2D square domain. The positions of solid-liquid interface at time (a) $t = 7$, (b) $t = 56$, (c) $t = 105$, and (d) $t = 210$ are shown. Solid and liquid particles are marked in red and blue colors, respectively.

Here, we have performed an averaging process based on 30 samples obtained from different random seeds. It is interesting to note that the number increases in an exponential manner and peaks at $t \sim 120$. Thereafter, the number of mushy particles decays steadily until a full solid is formed at $t \sim 325$.

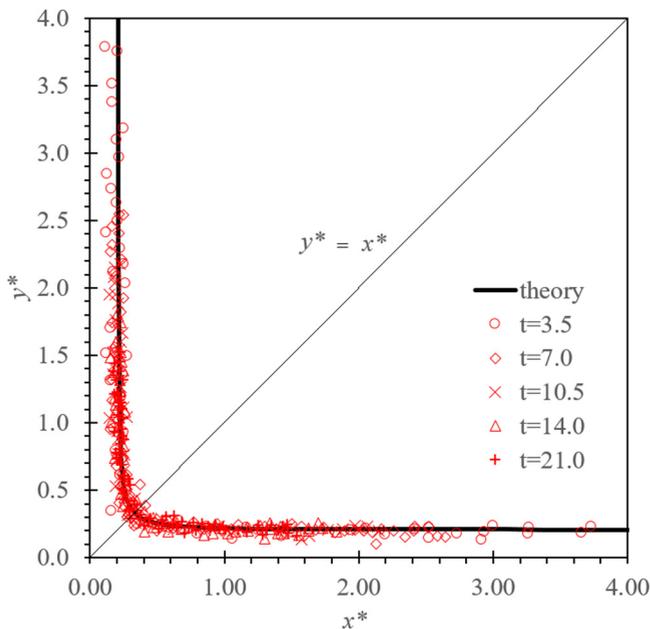


FIG. 12. Comparison of the instantaneous dimensionless positions of mushy particles (x^*, y^*) with the analytical solution at different time levels.

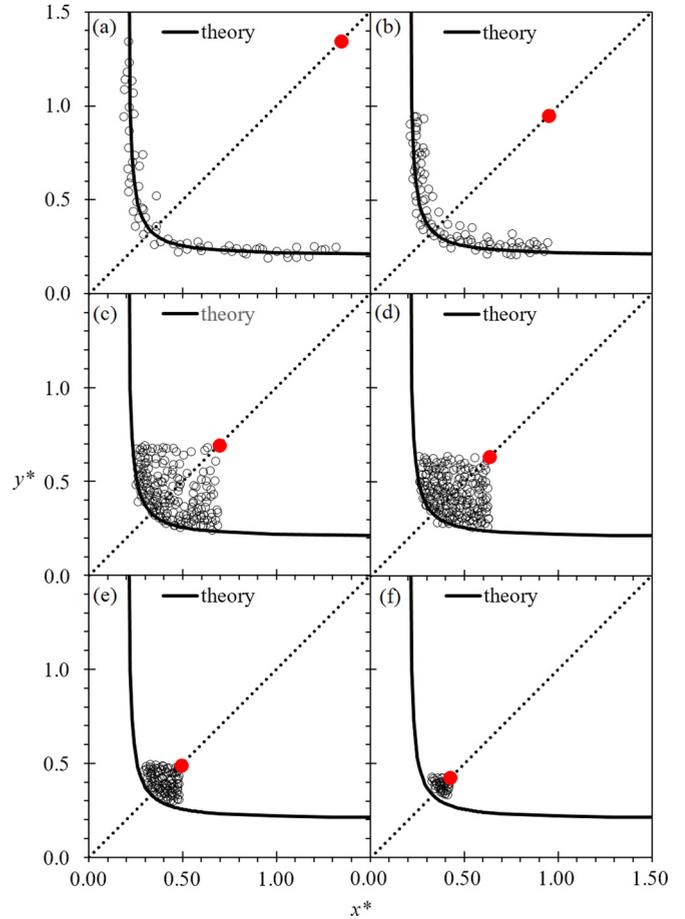


FIG. 13. Comparison of instantaneous dimensionless positions of mushy particles (x^*, y^*) with the analytical solution for (a) $t = 28$, (b) $t = 56$, (c) $t = 105$, (d) $t = 126$, (e) $t = 210$, and (f) $t = 280$. The linear dashed line indicates $y^* = x^*$. The red solid circle represents the dimensionless position of the square center. For these particular instants, the eDPD solutions do not agree with the analytical solutions due to the semi-infinite assumption made in the latter.

Finally, we wish to verify our model with the experimental results of 2D solidification performed earlier by Jiji *et al.* [67]. Similar to the experimental conditions, we have treated the top and right walls as adiabatic. The temperatures of the left and bottom walls were dropped below the freezing temperature in

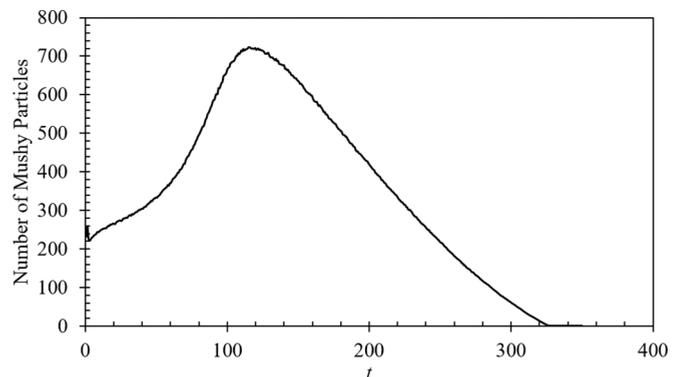


FIG. 14. Total number of mushy particles during the 2D solidification process.

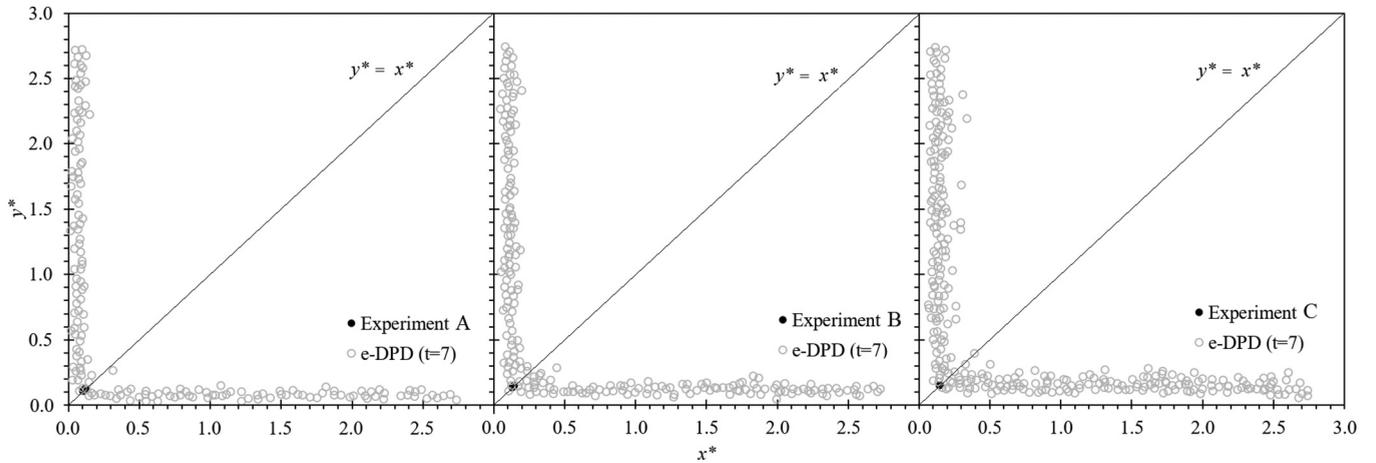


FIG. 15. Comparison of x_o^* values obtained from experiments (Jiji *et al.* [67]) and eDPD for three conditions. Experiment A: $St = 1/33.9$ and $T_o^* = 1.35(0.104 < x_o^* < 0.125)$; experiment B: $St = 1/22.4$ and $T_o^* = 0.553(0.134 < x_o^* < 0.146)$; experiment C: $St = 1/19.6$ and $T_o^* = 0.25(0.149 < x_o^* < 0.154)$.

order to initiate the solidification process. Figure 15 shows the experimental values of x_o^* at three experimental conditions (A, B, and C). Here, x_o^* is the interface location at $y^* = x^*$. As seen, the x_o^* values predicted from eDPD agree considerably well with those measured. Experimentally, we found that the averaged x_o^* increased (more rapid solidification) as $1/St$ and T_o^* decreased (i.e., experiment A \rightarrow C). A similar trend is observed in the present eDPD results.

V. CONCLUSIONS

In the current work, a more systematic method has been proposed to predict the temperature-dependent function of the exponent s appearing in the weight function of the dissipative force term. By estimating the critical cutoff radius ($r_{c,crit}$) from the kinetic theory, one can then select a cutoff radius (r_c) that satisfies $r_c > r_{c,crit}$ to somewhat reproduce the Schmidt number of a physical fluid. Although the thermophysical properties of water including Schmidt and Prandtl numbers have been well captured within the temperature range considered in the current work, $253.15 < T^R < 373.15$ K, it is important to note that not all physical properties can be simultaneously reproduced (e.g., dielectric property). Therefore, the use of eDPD fluid to exactly reproduce a physical fluid of interest remains as an open problem.

The method has been verified against the benchmark solutions of some classical solidification problems, whereby the analytical and the experimental data are available. In order to properly simulate the speed of the solid-liquid interface, mushy particles have been proposed to model the eDPD particles within the transition zone. The effective thermal

diffusivities of mushy particles have been linearly interpolated based on the computed solid fraction. We have found that this method is able to smoothen the unphysical temperature fluctuation near the solid-liquid interface, particularly when the thermal conductivity of a solid varies considerably from that of a liquid.

Also, we have attempted an alternative way of implementing the constant temperature boundary conditions, without updating the positions of ghost particles at every time step as suggested in the literature. Accordingly, we have generated the mirror particles (of the ghost particles in the wall layer) inside the flow domain only once. The temperatures of these mirror particles are then interpolated from the neighboring eDPD particles. We foresee that this method is more robust if a curved wall is encountered. It has been found that the near-wall temperatures have been represented well by using the current approach. Also, we have further verified the eDPD method in a 2D solidification problem. The eDPD solidification results have been compared with both theoretical and experimental data and good agreements have been found.

This method will be extended to simulate the freezing processes in microfluidic devices such as phase-change valves and PEMFC. The results will be reported in the near future.

ACKNOWLEDGMENTS

The current work was performed when the first author joined the National Center for Theoretical Sciences (NCTS), NTU, Taipei as a visiting scholar. He would like to acknowledge the financial assistance provided by the National Center for Theoretical Sciences (NCTS) and Universiti Tenaga Nasional (UNITEN) during his visit.

- [1] J. J. Monaghan, Smoothed particle hydrodynamics, *Annu. Rev. Astron. Astrophys.* **30**, 543 (1992).
 [2] A. Nakayama, L. Y. Leong, and W. S. Kong, Application of smoothed particle hydrodynamics with wall turbulence model to computation of high speed spillway flows, *Mem. Constr. Eng. Res. Inst.* **58**, 1 (2016).

- [3] A. Nakayama, L. Y. Leong, and W. S. Kong, Simulation of turbulent free surface flow with two-phase SPH method, *J. Jpn. Soc. Civil Eng.* **72**, I_523 (2016).
 [4] S. Koshizuka and Y. Oka, Moving particle semi-implicit method for fragmentation of incompressible fluid, *Nucl. Sci. Eng.* **123**, 421 (1996).

- [5] K. C. Ng, E. Y. K. Ng, and W. H. Lam, Lagrangian simulation of steady and unsteady laminar mixing by plate impeller in a cylindrical vessel, *Ind. Eng. Chem. Res.* **52**, 10004 (2013).
- [6] Y. H. Hwang, K. C. Ng, and T. W. H. Sheu, An improved particle smoothing procedure for Laplacian operator in a randomly scattered cloud, *Numer. Heat Transfer, Part B* **70**, 111 (2016).
- [7] K. C. Ng, T. W. H. Sheu, and Y. H. Hwang, On the extension of moving particle method for flow computation in irregular flow domain, *Pertanika J. Sci. Technol.* **24**, 465 (2016).
- [8] K. C. Ng, S. M. Yusoff, Y. H. Hwang, T. W. H. Sheu, and M. Z. Yusoff, Simulation of unsteady incompressible flow using a new particle method, *Appl. Mech. Mater.* **819**, 326 (2016).
- [9] K. C. Ng, Y. L. Ng, and W. H. Lam, Particle simulation and flow sequence on drainage of liquid particles, *Comput. Math. Appl.* **66**, 1437 (2013).
- [10] K. C. Ng and E. Y. K. Ng, Laminar mixing performance of baffling, shaft eccentricity and unsteady mixing in a cylindrical vessel, *Chem. Eng. Sci.* **104**, 960 (2013).
- [11] K. C. Ng, Y. H. Hwang, and T. W. H. Sheu, On the accuracy assessment of Laplacian models in MPS, *Comput. Phys. Commun.* **185**, 2412 (2014).
- [12] K. C. Ng, Y. H. Hwang, T. W. H. Sheu, and C. H. Yu, Moving particle level set (MPLS) method for incompressible multiphase flow computation, *Comput. Phys. Commun.* **196**, 317 (2014).
- [13] K. C. Ng and T. W. H. Sheu, On the Laplacian model for particle-based simulation using Moving-Particle Semi-implicit (MPS) Method, in *Proceedings of the Sixth International Conference on Machine Vision (ICMV 13)* (SPIE, Bellingham, WA, 2013), pp. 90672H–90672H.
- [14] K. C. Ng, Y. L. Ng, and M. Z. Yusoff, Development of a Lagrangian meshless flow solver based on the moving particle semi-implicit (MPS) method, *IOP Conf. Ser.: Earth Environ. Sci.* **16**, 012151 (2013).
- [15] Y. L. Ng, K. C. Ng, and M. Z. Yusoff, The study of pressure source term in moving particle semi-implicit (MPS), *IOP Conf. Ser.: Earth Environ. Sci.* **16**, 012152 (2013).
- [16] K. S. Liu, T. W. H. Sheu, Y. H. Hwang, and K. C. Ng, High-order particle method for solving incompressible Navier-Stokes equations within a mixed Lagrangian-Eulerian framework, *Comput. Methods Appl. Mech. Eng.* **325**, 77 (2017).
- [17] K. C. Ng, Y. H. Hwang, and T. W. H. Sheu, Recent progress on the development of a particle method for incompressible single- and multi-phase flow computation, *ARPN J. Eng. Appl. Sci.* **12**, 2358 (2017).
- [18] A. M. E. Ibrahim, K. C. Ng, and M. Z. Yusoff, Parallelization of the moving particle pressure mesh (MPPM) fluid flow solver by using OPENMP, *ARPN J. Eng. Appl. Sci.* **12**, 2515 (2017).
- [19] P. J. Hoogerbrugge and J. M. V. A. Koelman, Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics, *Europhys. Lett.* **19**, 155 (1992).
- [20] D. Duong-Hong, N. Phan-Thien, K. S. Yeo, and G. Ausias, Dissipative particle dynamics simulations for fibre suspensions in Newtonian and viscoelastic fluids, *Comput. Methods Appl. Mech. Eng.* **199**, 1593 (2010).
- [21] D. A. Fedosov, A. Sengupta, and G. Gompper, Effect of fluid-colloid interactions on the mobility of a thermophoretic microswimmer in non-ideal fluids, *Soft Matter* **11**, 6703 (2015).
- [22] D. F. Hinz, A. Panchenko, T. Y. Kim, and E. Fried, Particle-based simulations of self-motile suspensions, *Comput. Phys. Commun.* **196**, 45 (2015).
- [23] A. Kumar, Computational modeling of microfluidic processes using dissipative particle dynamics, Ph.D. thesis, University of Rhode Island, 2009.
- [24] J. Guo, X. Li, Y. Liu, and H. Liang, Flow-induced translocation of polymers through a fluidic channel: A dissipative particle dynamics simulation study, *J. Chem. Phys.* **134**, 134906 (2011).
- [25] Y. H. Tang, Z. Li, X. Li, M. Deng, and G. E. Karniadakis, Non-equilibrium dynamics of vesicles and micelles by self-assembly of block copolymers with double thermoresponsivity, *Macromolecules* **49**, 2895 (2016).
- [26] X. Fan, N. Phan-Thien, T. Y. Ng, X. Wu, and D. Xu, Microchannel flow of a macromolecular suspension, *Phys. Fluids* **15**, 11 (2003).
- [27] X. Fan, N. Phan-Thien, S. Chen, X. Wu, and T. Y. Ng, Simulating flow of DNA suspension using dissipative particle dynamics, *Phys. Fluids* **18**, 063102 (2006).
- [28] T. Steiner, C. Cupelli, and R. Zengerle, Simulation of advanced microfluidic systems with dissipative particle dynamics, *Microfluid. Nanofluid.* **7**, 307 (2009).
- [29] E. Moeendarbary, T. Y. Ng, and M. Zangeneh, Dissipative particle dynamics: Introduction, methodology, and complex fluid applications—a review, *Int. J. Appl. Mech.* **1**, 737 (2009).
- [30] Z. G. Mills, W. Mao, and A. Alexeev, Mesoscale modeling: Solving complex flows in biology and biotechnology, *Trends Biotechnol.* **31**, 426 (2013).
- [31] M. B. Liu, G. R. Liu, L. W. Zhou, and J. Z. Chang, Dissipative particle dynamics (DPD): An overview and recent developments, *Arch. Comput. Methods Eng.* **22**, 529 (2014).
- [32] P. Espanol and P. B. Warren, Perspective: Dissipative particle dynamics, *J. Chem. Phys.* **146**, 150901 (2017).
- [33] H. Lei, Dissipative particle dynamics: Theory, algorithms and application to sickle cell anemia, Ph.D. thesis, Brown University, 2012.
- [34] P. Espanol and M. Revenga, Smoothed dissipative particle hydrodynamics, *Phys. Rev. E* **67**, 026705 (2003).
- [35] P. Espanol, Dissipative particle dynamics with energy conservation, *Europhys. Lett.* **30**, 191 (1997).
- [36] J. B. Avalos and A. D. Mackie, Dissipative particle dynamics with energy conservation, *Europhys. Lett.* **40**, 141 (1997).
- [37] A. Chaudhri and J. R. Lukes, Multicomponent energy conserving dissipative particle dynamics: A general framework for mesoscopic heat transfer applications, *J. Heat Transfer* **131**, 033108 (2009).
- [38] M. Ripoll, P. Espanol, and M. H. Ernst, Dissipative particle dynamics with energy conservation: Heat conduction, *Int. J. Mod. Phys. C* **9**, 1329 (1998).
- [39] A. D. Mackie, J. B. Avalos, and V. Navas, Dissipative particle dynamics with energy conservation: Modelling of heat flow, *Phys. Chem. Chem. Phys.* **1**, 2039 (1999).
- [40] Y. X. Zhang, H. L. Yi, and H. P. Tan, A dissipative particle dynamics algorithm for fluid-solid conjugate heat transfer, *Int. J. Heat Mass Transfer* **103**, 555 (2016).
- [41] R. Qiao and P. He, Simulation of heat conduction in nanocomposite using energy-conserving dissipative particle dynamics, *Mol. Simul.* **33**, 677 (2007).
- [42] S. M. Willemsen, H. C. J. Hoefsloot, D. C. Visser, P. J. Hamersma, and P. D. Iedema, Modelling phase change with dissipative particle dynamics using a consistent boundary condition, *J. Comput. Phys.* **162**, 385 (2000).

- [43] S. M. Willemsen, H. C. J. Hoefsloot, and P. D. Iedema, Mesoscopic simulation of polymers in fluid dynamics problems, *J. Stat. Phys.* **107**, 53 (2002).
- [44] E. O. Johansson, T. Yamada, B. Sundén, and J. Yuan, Modeling mesoscopic solidification using dissipative particle dynamics, *Int. J. Therm. Sci.* **101**, 207 (2016).
- [45] Z. Li, Y. H. Tang, H. Lei, B. Caswell, and G. E. Karniadakis, Energy-conserving dissipative particle dynamics with temperature-dependent properties, *J. Comput. Phys.* **265**, 113 (2014).
- [46] E. Abu-Nada, Dissipative particle dynamics simulation of natural convection using variable thermal properties, *Int. Commun. Heat Mass Transfer* **69**, 84 (2015).
- [47] T. Yamada, E. O. Johansson, B. Sundén, and J. Yuan, Dissipative particle dynamics simulations of water droplet flows in a submicron parallel-plate channel for different temperature and surface-wetting conditions, *Numer. Heat Transfer, Part A* **70**, 595 (2016).
- [48] R. D. Groot and P. B. Warren, Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation, *J. Chem. Phys.* **107**, 4423 (1997).
- [49] G. E. Fabritiis, M. Serrano, P. Espanol, and P. V. Coveney, Efficient numerical integrators for stochastic models, *Phys. A (Amsterdam, Neth.)* **361**, 429 (2006).
- [50] Z. Rao, Y. Huo, and X. Liu, Dissipative particle dynamics and experimental study of alkane-based nanoencapsulated phase change material for thermal energy storage, *RSC Adv.* **4**, 20797 (2014).
- [51] N. Arai, K. Yasuoka, and X. C. Zeng, A vesicle cell under collision with a Janus or homogeneous nanoparticle: Translocation dynamics and late-stage morphology, *Nanoscale* **5**, 9089 (2013).
- [52] J. B. Avalos and A. D. Mackie, Dynamic and transport properties of dissipative particle dynamics with energy conservation, *J. Chem. Phys.* **111**, 5267 (1999).
- [53] N. Mai-Duy, D. Pan, N. Phan-Thien, and B. C. Khoo, Dissipative particle dynamics modeling of low Reynolds number incompressible flows, *J. Rheol.* **57**, 585 (2013).
- [54] Z. Li and G. Drazer, Hydrodynamics interactions in dissipative particle dynamics, *Phys. Fluids* **20**, 103601 (2008).
- [55] M. Ripoll, K. Mussawisade, R. G. Winkler, and G. Gompper, Dynamic regimes of fluids simulated by multiparticle-collision dynamics, *Phys. Rev. E* **72**, 016701 (2005).
- [56] V. Symeonidis, G. E. Karniadakis, and B. Caswell, Schmidt number effects in dissipative particle dynamics simulation of polymers, *J. Chem. Phys.* **125**, 184902 (2006).
- [57] S. S. Dhanabalan, On the boundary conditions for dissipative particle dynamics, Master's thesis, National University of Singapore, 2005.
- [58] V. Holtén, C. E. Bertrand, M. A. Anisimov, and J. V. Sengers, Thermodynamics of supercooled water, *J. Chem. Phys.* **136**, 094507 (2012).
- [59] J. Hallett, The temperature dependence of the viscosity of supercooled water, *Proc. Phys. Soc.* **82**, 1046 (1963).
- [60] K. T. Gillen, D. C. Douglass, and M. J. R. Hoch, Self-diffusion in liquid water to -31°C , *J. Chem. Phys.* **57**, 5117 (1972).
- [61] J. A. Backer, C. P. Lowe, H. C. Hoefsloot, and P. D. Iedema, Poiseuille flow to measure the viscosity of particle model fluids, *J. Chem. Phys.* **122**, 154503 (2005).
- [62] V. J. Lunardini, in *Heat Transfer with Freezing and Thawing*, Developments in Geotechnical Engineering (Elsevier Science, Amsterdam, 1991), Vol. 65, p. 1–437.
- [63] V. Alexiades and A. D. Solomon, *Mathematical Modeling of Melting and Freezing Processes* (Hemisphere Publishing Corporation, Washington, DC, 1993).
- [64] R. S. Mahmudah, M. Kumabe, T. Suzuki, L. Guo, and K. Morita, Particle-based simulations of molten metal flows with solidification, *Mem. Fac. Eng., Kyushu Univ.* **71**, 17 (2011).
- [65] K. C. Ng, T. W. H. Sheu, and Y. H. Hwang, Unstructured moving particle pressure mesh (UMPPM) method for incompressible isothermal and non-isothermal flow computation, *Comput. Methods Appl. Mech. Eng.* **305**, 703 (2016).
- [66] K. A. Rathjen and L. M. Jiji, Heat conduction with melting or freezing in a corner, *J. Heat Transfer* **93**, 101 (1971).
- [67] L. M. Jiji, K. A. Rathjen, and T. Drzewiecki, Two-dimensional solidification in a corner, *Int. J. Heat Mass Transfer* **13**, 215 (1970).