Modelling Solidification Problem Using Energy-Conserving Dissipative Particle Dynamics

K.C. Ng, NCTS
T.W.H. Sheu, NTU
Y. L. Ng, NTU

NCTS/Math Technical Report
2017-001

National Center for
Theoretical Sciences
Mathematics Division, Taiwan

Modelling Solidification Problem using Energy-conserving Dissipative Particle Dynamics

K.C. Ng
National Center for Theoretical Sciences (NCTS), National Taiwan University, Taipei, Taiwan.
Department of Mechanical Engineering, Universiti Tenaga Nasional, Jalan IKRAM-UNITEN, 43000 Kajang, Selangor, Malaysia.
Email: ngkhaiching2000@yahoo.com
Tel: +60389212020 (Ext: 6478)

T.W.H. Sheu
Center for Advanced Study on Theoretical Sciences (CASTS), National Taiwan University, Taipei, Taiwan.
Email: twhsheu@ntu.edu.tw

Y. L. Ng
Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, Taiwan.
Email: d03525004@ntu.edu.tw

Abstract

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 a temperature-dependent weight function appeared in the dissipative force term. In this paper, a simple method used to devise the temperature-dependent weight function is proposed and the systematic approach in selecting the eDPD parameter such as the cut-off radius (r_c) is outlined. The method is then used to study a variety of phase-change problems involving solidification. The concept of "mushy" eDPD particle is 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 new way to implement the constant temperature boundary condition at the wall is proposed. The numerical solutions of the 1D and 2D solidification problems are then compared with the analytical solutions and it is found that the agreements are promising. Finally, the fluid flow and freezing behaviors of water in a channel is studied.

Keywords: DPD; solidification; CFD; particle method; mushy zone

1.0 Introduction

Dissipative Particle Dynamics (DPD) is a particle-based method originally proposed by Hoogerbrugge and Koelman (1992). In general, the particle considered in DPD represents a cluster of atoms/molecules (coarse-grained), instead of individual atom/molecule considered in Molecular Dynamics (MD). Therefore, DPD can be used to compute problems of larger spatial and time scales than those considered in MD such as fiber suspension (Duong-Hong et al. 2010), microswimmer (Fedosov et al. 2015; Hinz et al. 2015), blood flow (Kumar 2009), polymer dynamics (Guo et al. 2011; Tang et al. 2016), DNA suspensions (Fan et al. 2003, 2006), microfluidic systems (Steiner et al. 2009), etc. A more thorough review on the applications of DPD can be found in the works reported by Moeendarbary et al. (2009), Mills et al. (2013) and Liu et al. (2014).

In order to extend the numerical framework of DPD to handle non-isothermal flow, Espanol (1997) has developed the transport equation of internal energy for each DPD particle. Espanol's method is commonly known as the energy-conserving Dissipative Particle Dynamics (eDPD) in open literature. More recently, Chaudhri and Lukes (2009) have extended the formulations of eDPD method to compute multicomponent flow. To date, eDPD has been used in many fluid flow applications involving heat transfer such as conjugate heat transfer problem (Zhang et al. 2016), conduction problem in nanocomposite (Qiao and He 2007), thermophoretic microswimmer problem (Fedosov et al. 2015), etc. By combining the enthalpy method with eDPD, problems involving phase-change (i.e. melting) have been addressed by Willemsen et al. (2000, 2002).

It is well-known that thermophysical property of a real fluid changes with respect to temperature. In fact, for common liquid such as water, its dynamic viscosity decreases as temperature increases. Li et al. (2014) have recently devised a temperature-dependent formulation for the exponent s (denoted as Li's formulation in this paper) appeared in the weight function of dissipative force in order to capture this physical fluid property by using eDPD. Basically, they have enlarged the cut-off 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 successfully reproduced the Schmidt number and the Prandtl number for liquid water at various temperatures. In their works, they have recommended the values of C_1 and C_2 as 0.41 and 1.9, respectively. However, the details on how to choose the values of r_c, C_1 and C_2 are not given. More recently, Abu-Nada (2015) has argued on the difficulty in extending the Li's formulation to other fluids. In order to resolve this issue, Abu-Nada (2015) has incorporated the temperature-dependent viscosity ratio in the w_D weight function of Fan et al. (2006). Although the extension of this method to handle other fluids is straightforward, it remains as open question whether this method is able to reproduce the Schmidt number of a real fluid at different temperatures. Yamada et al. (2016) have recently highlighted the importance of modelling the freezing behavior in microfluidic devices such as phase change valves and Proton Exchange Membrane Fuel Cells (PEMFC) in order to control the ice formation. Following this, Johansson et al. (2016) have adopted the Li's formulation in modelling the freezing behavior in a flow channel. They have used a larger r_c in their 2D eDPD simulation (r_c =1.81); however, the rationale behind this selection is unclear.

In the current paper, we intend to propose a new formulation of the exponent s. The selection of the proper cut-off 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 for water below 0^{o}C (i.e. supercooled water) is available, we have further extended this method to cover a wider temperature range, i.e. 253.15 K < T^R < 373.15K, where T^R is the temperature with physical unit. The new formulation is 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. (2000) by simply fixing the ghost particles in the wall and their temperatures are determined based on the numerical interpolation in the fluid region. Therefore, it is unnecessary to update the positions of ghost particles in our current implementation. We have further validated the eDPD method in 2D solidification problem (where analytical solution is available) which has been overlooked by the eDPD researchers so far. Finally, the fluid flow and freezing behaviors in a flow channel will be studied.

2.0 Mathematical Model

The governing equations of energy-conserving Dissipative Particle Dynamics (eDPD) model have been reported in many literatures. Here, these equations are rewritten for completeness. The equation of motion of an eDPD particle *i* can be expressed as:

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

$$\frac{d\vec{r}_i}{dt} = \vec{v}_i \, , \tag{2}$$

where $\vec{F}_{ij}^{C}$ is the conservative force vector, consisting of soft repulsive force due to the interaction potential between particles and $\vec{F}_{ij}^{D}$ is the dissipative force vector which would dissipate the thermal kinetic energy of the system and hence reduce the velocity difference between the particles. On the other hand, the random force vector, $\vec{F}_{ij}^{R}$ generates a stochastic force on 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 are having the same mass *m* (=1.0), and the velocity vector $\vec{v}_i$ is found by integrating Equations (1, 2) with respect to time by using the velocity-Verlet scheme proposed by Groot and Warren (1997). The summation operator appeared in Equation (1) is performed on all neighbouring eDPD particles within the cut-off radius r_c. Mathematically, these force components are expressed as:

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

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

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

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 $a_{ij} = 75 k_B T_{ij} / \rho$ (for water) where k_B is the Boltzmann constant. T_{ij} is the effective temperature and ρ is the number density. Li et al. (2014) have defined $T_{ij} = (T_i + T_j)/2$ in their work. However, we follow the approach of Fedosov et al. (2015) by using a constant value of T_{ij}, i.e. $T_{ij} = (T_H + T_C)/2$ where T_H and T_C are the hot and cold temperatures in the system. γ_{ij} and σ_{ij} are the dissipative and random force parameters, respectively, satisfying the fluctuation-dissipation theorem: $\sigma_{ij}^2 = 4\gamma_{ij} k_B T_i T_j / (T_i + T_j)$ and $w_D = w_R^2$. The weight functions such as $w_C = 1 - r/r_c$ and $w_D = (1 - r/r_c)^s$ (where $s = 2$) are commonly adopted. ξ_{ij} is the symmetric random number with zero mean and unit variance, i.e. $\xi_{ij} = \xi_{ji}$. Qiao and He (2007) have previously reported that result simulated based on uniform random number is similar to that of Gaussian random number. Therefore, due to the fact that the generation of uniform random number is computationally cheaper, it is implemented in the current work.

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

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

where e_i is the internal energy of particle *i*, i.e. $e_i = C_v T_i$ where 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 q_{ij}^R are written as:

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

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

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

External heat source is included in Q_{ext}. The term k_{ij} indicates the strength of the collisional heat flux, which is related to the mesoscopic heat friction κ_{ij} as $k_{ij} = C_v^2 \kappa_{ij} (T_i + T_j)^2 / 4k_B$. As reported by Zhang et al. (2016), the mesoscopic heat friction κ_{ij} is treated as heat conductivity in mesoscopic scale; therefore, the harmonic mean procedure is used to express the equivalent heat friction between 2 particles:

$\kappa_{ij} = 2\kappa_i \kappa_j /(\kappa_i + \kappa_j)$. 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 $w_{RT} = (1 - r/r_c)$ are used. ξ_{ij}^e is the anti-symmetric 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 are written in their dimensionless forms. In the current work, the reference temperature T_*^R is 300K. Here, the superscript *R* signifies the quantity with physical units and the subscript * indicates the reference quantity. By normalizing the specific heat of a reference mass ($C_v^R m_*^R$) with the Boltzmann constant (k_B^R=1.381x10^{-23}J/K), one could obtain the dimensionless specific heat C_v defined earlier, i.e. $C_v = C_v^R m_*^R / k_B^R$, where m_*^R is the mass of an eDPD particle (reference mass) which can be written as $m_*^R = \rho^R \left(l_*^R\right)^3 / \rho$. Here, a physical space of volume $\left(l_*^R\right)^3$, where l_*^R is the reference length scale, contains ρ number of eDPD particles. Therefore, the total specific heat within this physical space, i.e. $\rho C_v k_B^R$ is related to the real specific heat as: $\rho C_v k_B^R = \rho^R \left(l_*^R\right)^3 C_v^R$. By treating the density of water ρ^R as 1000kg/m^3 and the specific heat of water C_v^R as 4180 J/kgK and using the standard eDPD parameters such as ρ = 4 and C_v =100,000, one can compute the length scale as l_*^R ~ 11nm. In general, the eDPD model is more coarse-grained when C_v increases.

3.0 Thermophysical properties of water

The exponent *s* appeared 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 real fluid, i.e. Schmidt number (*Sc*) is ~1.0 (Groot and Warren 1997). However, for real fluid such as water, *Sc* ~ 350 at T_*^R = 300K, indicating that the simulated *Sc* would be several orders of magnitude lower than the actual *Sc* by using the standard DPD parameters. This condition occurs due to several reasons. Apart from the soft interaction between the DPD particles (Mai-Duy et al. 2013), shearing dissipation may disappear even though two DPD particles are close to each other (Fan et al. 2006). In general, fluid is behaving like gas if *Sc* is low or liquid if *Sc* is large (Li and Drazer 2008). Ripoll et al. (2005) have observed that the correct hydrodynamic behavior of suspended particles can be recovered only at large *Sc*. Symeonidis et al. (2006) have compared the DPD results with the experimental data and concluded that the under-predicted value of *Sc* would degrade the accuracy of the DPD simulation.

From the linearized Fokker-Planck equation (see Dhanabalan (2005)), the hydrodynamic variable such as kinematic viscosity υ and self-diffusivity *D* can be written as:

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

$$D = \frac{k_B T}{m w_0} \tag{11}$$

Here, $w_o = \gamma\rho[w_D]_r / md$ 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 cut-off 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 (υ) and self-diffusivity (*D*) can now be determined as a function of *s*. Here, we focus on two-dimensional problem (*d* = 2):

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

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

Thus, the Schmidt number can be written as:

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

From Equation (14), it is now straightforward to devise methods that are able to increase the Schmidt number of the DPD fluid. One may decrease the temperature *T* as practiced in Li and Drazer (2008). Alternatively, Fan et al. (2006) 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 parameter γ can be pursued; however, the random parameter σ must be increased (hence reduced Δt) to satisfy the fluctuation-dissipation theorem.

For non-isothermal 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. (2014) 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 273K < T^R < 373K. Meanwhile, they have increased the cut-off 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 (2015) has 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. (2006), 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. (2014), Johansson et al. (2016) have computed the 2D mesoscopic solidification problem using eDPD. They have enlarged the cut-off 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. (2014) and Johansson et al. (2016) at different temperatures are shown in Figure 1. As seen, the computed Schmidt numbers based on the *s*(*T*) function of Li et al. (2014) vary considerably with those measured experimentally, especially at low temperature regime. On the other hand, the over-prediction of the Schmidt numbers simulated by Johansson et al. (2006) is quite discernible in medium temperature range: 291K < T^R< 312K (or 0.97 < T < 1.04).

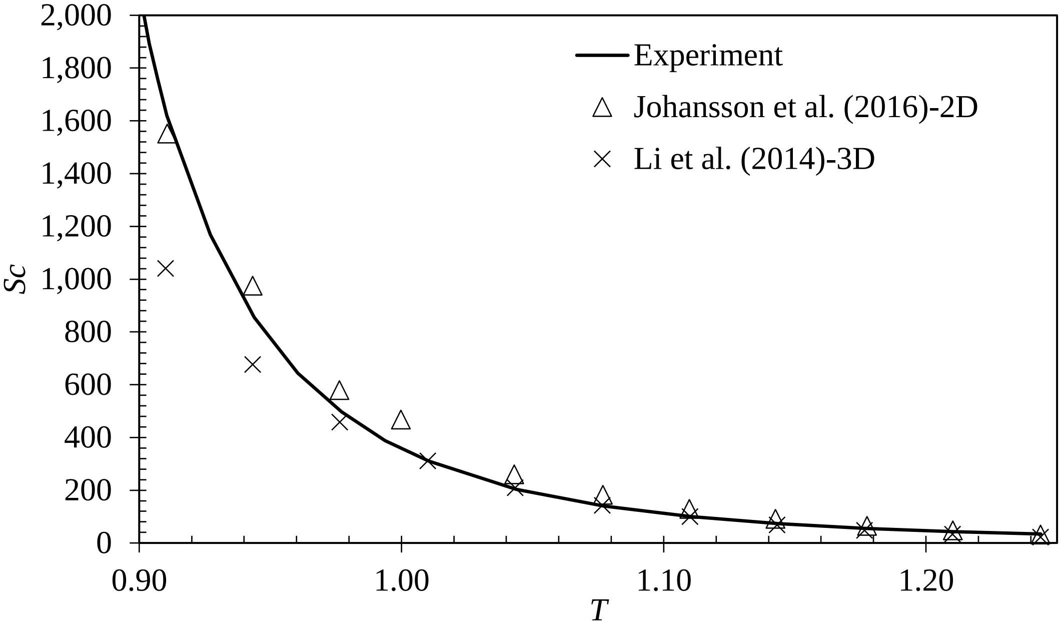


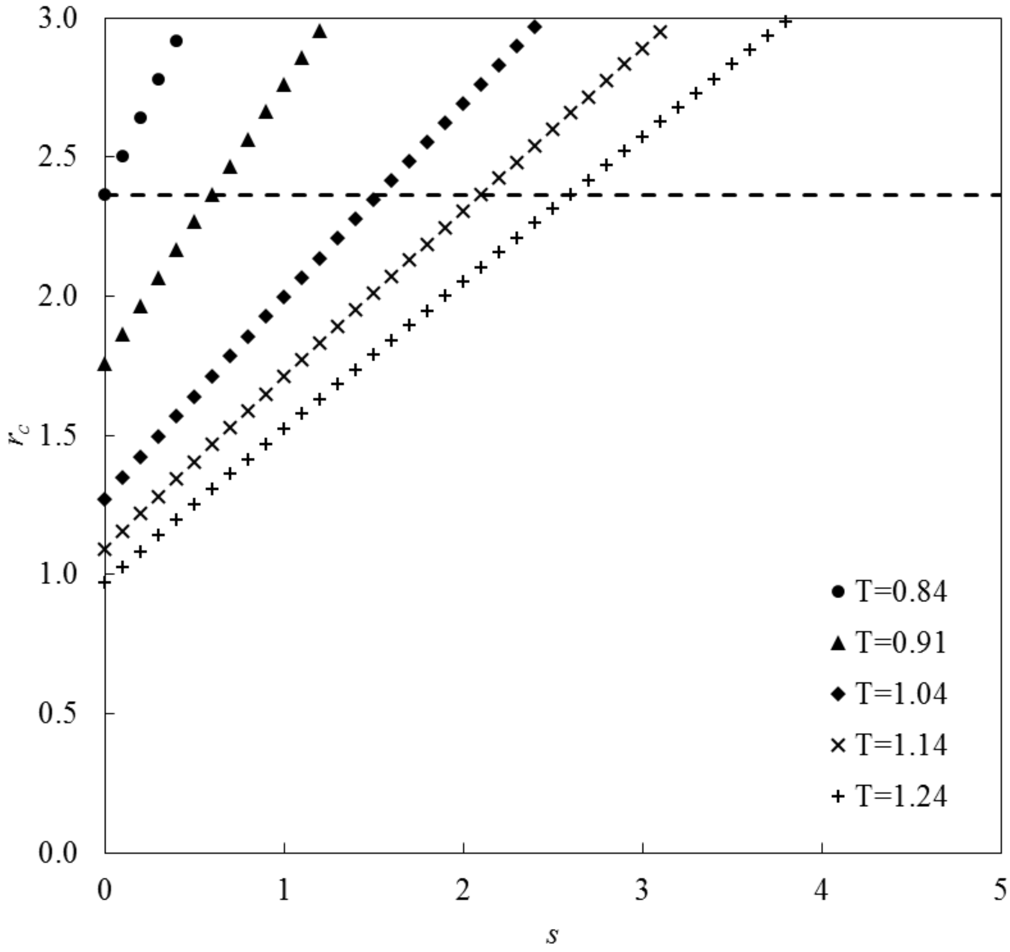
Figure 1: Schmidt number as a function of temperature predicted by Li et al. (2014) and Johansson et al. (2016).

Now, the main question is lying on how to choose the proper DPD parameters in order to reproduce the Schmidt number of water. Firstly, we realize that the cut-off radius r_c is a constant parameter (fixed during the eDPD computation). Therefore, Equation (14) can be rearranged in a way leading to:

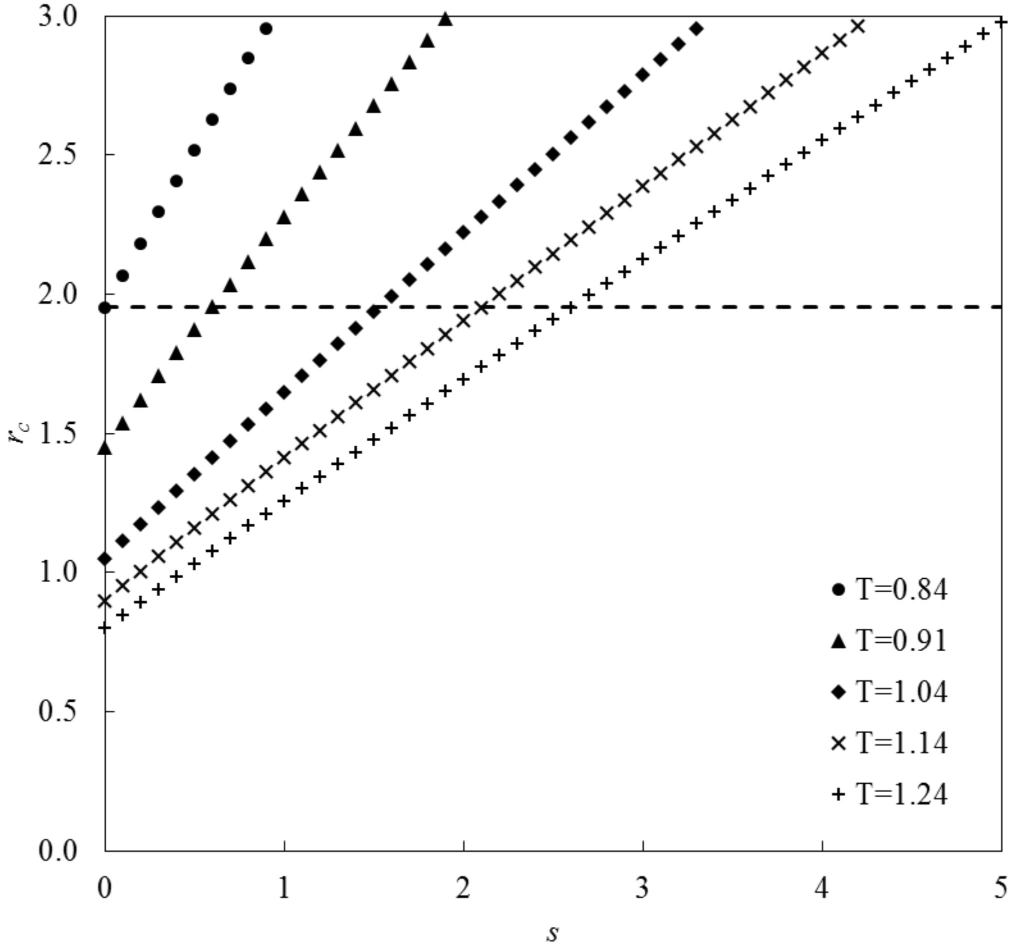
$$r_c = \left(\frac{\left(Sc - \frac{1}{2} \right) 4 k_B T (s+1)^2 (s+2)^2 (s+3)(s+4)}{3\pi^2 \gamma^2 \rho^2} \right)^{1/6} . \tag{15}$$

Now, we are able to generate plots of r_c against *s* for different temperatures as shown in Figure 2. Here, we replace the term *Sc* by the target value (i.e. experimental values of Schmidt number), and utilize the standard DPD parameters such as γ=4.5 and ρ=4 to generate the plots. It is important to note that the temperature range considered is subjected to the availability of the experimental value of Sc.

Due to the fact that the thermophysical properties of supercooled liquid water are experimentally available (see Holten et al. 2012, Hallett 1963 and Gillen et al. 1972), let us consider a working temperature range: $253.15 \text{ K} < T^R < 373.15\text{K}$ (or $0.84 < T < 1.24$), which is larger than that considered by Li et al. (2014). As observed from Figure 2(a), the exponent s is increasing with respect to temperature upon prescribing a constant r_c value, which is agreeable to the original proposal of Li et al. (2014). 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 Fan et al. 2006). As shown in Figure 2(a), in order to ensure $s>0$ (i.e. w_D is decaying within r_c) for the entire temperature range, the condition of $r_c > r_{c,crit}$ must hold. From Figure 2(a), it seems that $r_{c,crit} \sim 2.365$. Alternatively, the term $r_{c,crit}$ can be calculated from Equation (15) by setting $s = 0$ and $Sc = 10356$ (i.e. the experimental value of Schmidt number at the lowest temperature within the range considered: $T = 0.84$).



(a)



(b)

Figure 2: The relationship between r_c and s at different temperatures. Dash line indicates $r_c = r_{c,crit}$. (a) γ=4.5 and ρ=4 are used. $r_{c,crit}$ ~ 2.365. (b) γ=8.0 and ρ=4 are used. $r_{c,crit}$ ~ 1.952. r_c must be chosen in such a way that $r_c > r_{c,crit}$.

Now, if one relies on the standard parameters, e.g. γ=4.5 and ρ=4, one may choose r_c = 2.37 (i.e. a value which is slightly larger than $r_{c,crit}$ ~ 2.365) for the eDPD computation. However, this implies that particle information in at least 3 (the next integer of $r_{c,crit}$) neighbouring 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 parameter γ to 8.0 and retaining ρ=4, $r_{c,crit}$ can be reduced to 1.952 according to Figure 2(b) or Equation (15). Correspondingly, due to the increase of γ (hence increase of random parameter σ), the time step size is reduced from the recommended value Δt = 0.01 (Johansson et al. 2016) to Δt = 0.007 to ensure numerical stability. Therefore, the eDPD parameters, i.e. r_c = 1.96, γ = 8.0, ρ=4 and Δt = 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 (say from Figure 2(b)) or based on Equation (15). We have opted to solve Equation (15) numerically for s. Given the experimental value of Schmidt number Sc at a particular temperature T, the exponent s is solved iteratively by using the eDPD parameters suggested above. The numerical values of $s(T)$ are tabulated in Table 1. A quadratic function $s^*(T) = -6.3039T^2 + 19.594T - 12.032$ can be used to fit the $s(T)$ data as well with the coefficient of determination $R^2 = 0.9997$.

Table 1: Computed s values at different temperatures from Equation (15) based on r_c = 1.96, γ = 8.0 and ρ=4.

T^R(°C)	T^R (K)	T	Sc_{exp}	s	s^*
-20	253.15	0.8438	10355.66	0.0068	0.0133
-19	254.15	0.8472	9287.15	0.0366	0.0431
-18	255.15	0.8505	8406.42	0.0643	0.0728
-17	256.15	0.8538	7590.42	0.0933	0.1023
-16	257.15	0.8572	6867.35	0.1223	0.1316
-15	258.15	0.8605	6221.72	0.1514	0.1608
-14	259.15	0.8638	5620.84	0.1820	0.1899
-13	260.15	0.8672	5117.06	0.2107	0.2189
-12	261.15	0.8705	4667.65	0.2393	0.2477
-11	262.15	0.8738	4267.43	0.2677	0.2763
-10	263.15	0.8772	3902.49	0.2965	0.3048
-9	264.15	0.8805	3553.05	0.3273	0.3332
-8	265.15	0.8838	3224.68	0.3597	0.3615
-7	266.15	0.8872	2927.04	0.3928	0.3896
-6	267.15	0.8905	2662.79	0.4256	0.4175
-5	268.15	0.8938	2434.85	0.4571	0.4453
-4	269.15	0.8972	2235.97	0.4875	0.4730
-3	270.15	0.9005	2067.09	0.5159	0.5006
-2	271.15	0.9038	1893.11	0.5483	0.5280
-1	272.15	0.9072	1755.51	0.5763	0.5552
0	273.15	0.9105	1619.74	0.6067	0.5823
5	278.15	0.9272	1167.94	0.7337	0.7158
10	283.15	0.9438	854.60	0.8619	0.8458
15	288.15	0.9605	644.16	0.9834	0.9723
20	293.15	0.9772	497.08	1.0997	1.0953
25	298.15	0.9938	388.20	1.2153	1.2148
30	303.15	1.0105	309.44	1.3251	1.3308
40	313.15	1.0438	203.13	1.5395	1.5522
50	323.15	1.0772	139.81	1.7414	1.7597
60	333.15	1.1105	99.78	1.9335	1.9531
70	343.15	1.1438	73.52	2.1158	2.1325

80	353.15	1.1772	55.53	2.2906	2.2980
90	363.15	1.2105	43.11	2.4541	2.4494
100	373.15	1.2438	33.94	2.6146	2.5868

Note: s^* is obtained from the fitting quadratic function: $s^*(T) = -6.3039T^2 + 19.594T - 12.032$. $R^2 = 0.9997$.

In what follows, the Schmidt and Prandtl numbers of the eDPD fluid will be measured and compared with the experimental data.

3.1 Numerical measurement of self-diffusivity

Following Kumar (2009), the self-diffusivity of eDPD fluid can be measured by using the mean square displacement (χ):

$$\chi = \frac{\sum_{i=1}^{N}\left[\vec{r}_i(t) - \vec{r}_i(0)\right]^2}{N}. \tag{16}$$

The self-diffusivity D is related to χ via:

$$D = \lim_{t\to\infty}\frac{1}{(2d)t}\chi \tag{17}$$

Here, d is set to 2.0 for 2D problem and N is the total number of eDPD particles.

The computational domain used to measure D is a periodic box of size 30x30. Given an initial temperature T, the eDPD computation is executed for 140000 time steps to ensure thermal equilibrium. Then, the mean square displacement χ is measured for another 140000 time steps. Figure 3 compares the self-diffusivities at different temperatures. Here, the mean square displacement (χ) is scaled with $2d$ and the slope is indeed the self-diffusivity of the fluid. As expected, the self-diffusivity increases with respect to temperature.

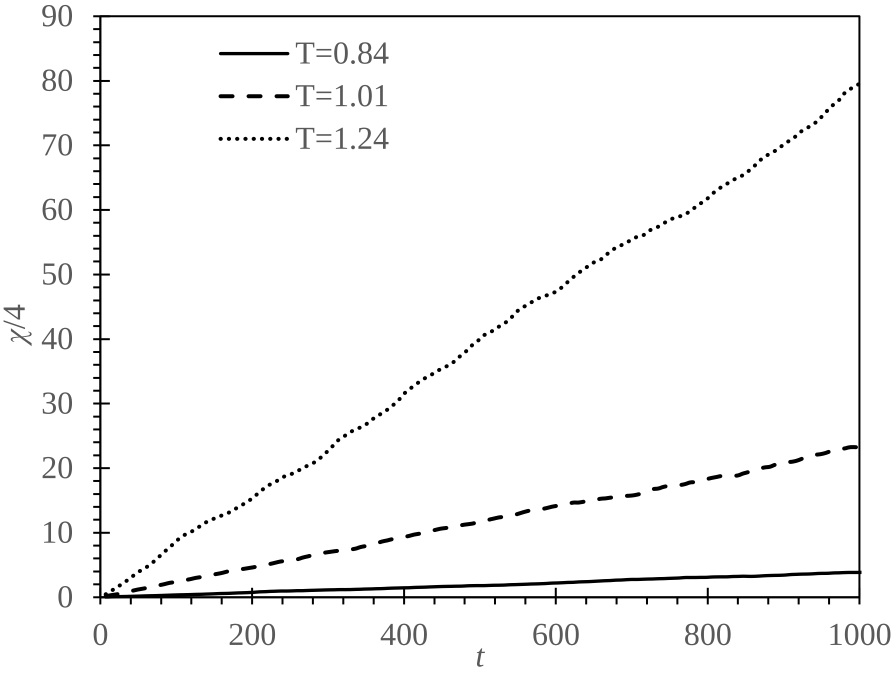


Figure 3: The mean squared displacement (scaled by 4) predicted at different temperatures. Self-diffusivity (slope) increases with respect to temperature.

3.2 Numerical measurement of kinematic viscosity and thermal diffusivity

In eDPD, the kinematic viscosity of the simulated fluid is an output parameter based on the specified eDPD parameters. Here, the Periodic Poiseuille Flow (PPF) method of Backer et al. (2005) has been used to determine the kinematic viscosity of the eDPD fluid. Figure 4 shows the schematic diagram of the PPF problem. The lengths of the periodic box in *x*- and *y*-directions (i.e. L_x and L_y) are set to 20 and 30, respectively. Depending on the flow region, the external force vector $\vec{F}^E$ is prescribed as $<\pm F_{ext}$, 0.0>, where $F_{ext} = 0.20$ (see Figure 4).

Firstly, the eDPD particles of initial temperature T are allowed to equilibrate for 100000 time steps (till t=7000). Then, each cell is divided into 40 bins in the *y*-direction and the time-averaging process is performed for another 120000 time steps. The time-averaged *x*-velocities (u) for different temperatures are shown in Figure 5. Based on the obtained maximum u (i.e. u_{max}) from the fitted parabolas, the kinematic viscosity υ at a particular temperature T can be computed as:

$$\nu = \frac{F_{ext} L_c^2}{2u_{\max}} \tag{18}$$

where L_c is the characteristic length: $L_c = L_y/4 = 7.5$ for the current PPF problem. In general, u_{max} declines in a more viscous medium. As observed from Figure 5, u_{max} drops (hence increasing viscosity) as T decreases, which is consistent with the physics of simple fluid such as water.

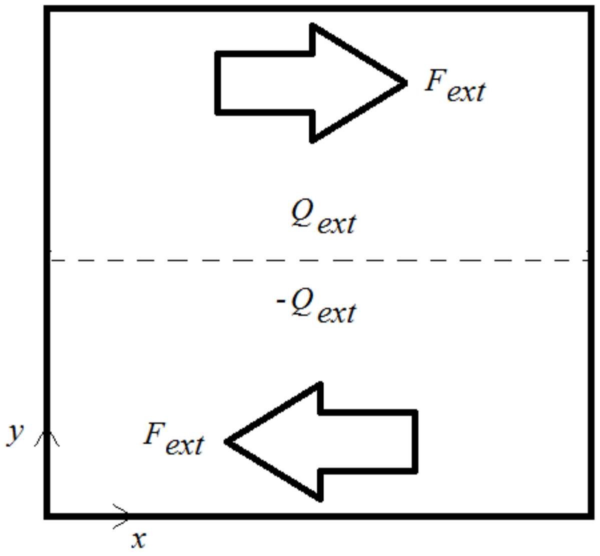


Figure 4: Schematic diagram of Periodic Poiseuille Flow (PPF) and Periodic Heat Conduction (PHC) problems. For PPF, F_{ext} = 0.2 and Q_{ext} = 0. For PHC, F_{ext} = 0 and Q_{ext} = 100. The dashed line is an imaginary line that divides the domain into two equal halves. Total lengths in x-direction (L_x) and y-direction (L_y) are 20 and 30, respectively. Periodic boundary conditions are applied at all boundaries.

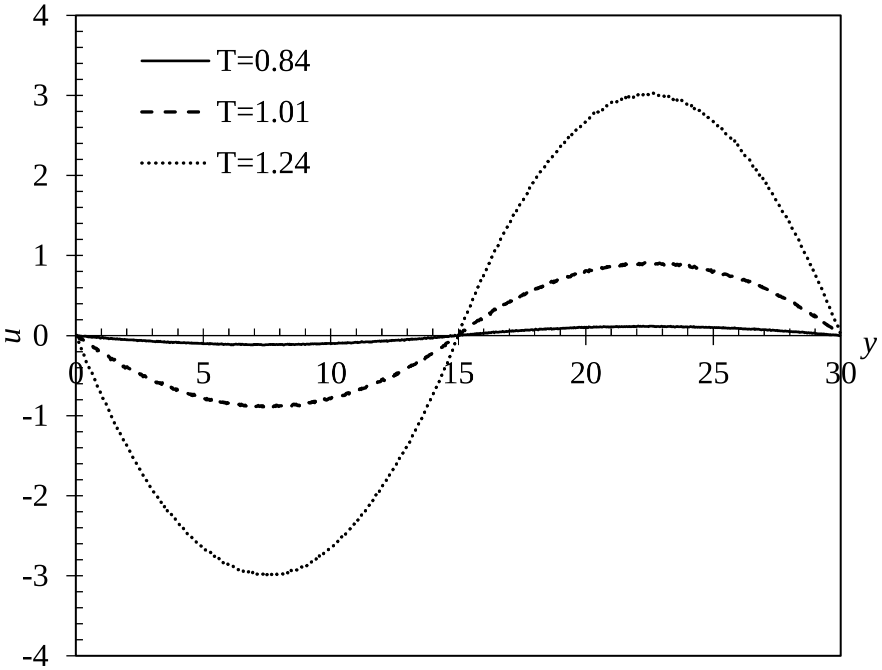


Figure 5: The velocity profiles of the Periodic Poiseuille Flow (PPF) problems for three temperatures (T=0.84, T=1.01 and T=1.24). The velocity profile is flattening as temperature decreases.

Recently, Li et al. (2014) have related the fluid property, i.e. Prandtl number (*Pr*) which is available experimentally to the mesoscopic heat friction κ. Based on the kinetic theory, α can be determined analytically as:

$$\alpha = \frac{\rho C_v \kappa}{2d}[r^2 w_{CT}]_r \tag{19}$$

By using the definition of Prandtl number ($Pr = \upsilon/\alpha$), the mesoscopic heat friction κ can be expressed as:

$$\begin{aligned}\kappa &= \frac{120}{\pi r_c^4 C_v \rho}\alpha \\ &= \frac{120}{\pi r_c^4 C_v \rho}\frac{\upsilon}{\Pr}\end{aligned}. \tag{20}$$

Here, Pr is the fluid property (dimensionless number) which is dependent on the temperature. Following the recommendations of Li et al. (2014), we have evaluated υ in the mesoscopic heat friction term analytically from Equation (12). The Prandtl number (Pr), however, is obtained directly from the experimental data.

Equation (20) serves as a method to estimate thermal diffusivity α from κ (i.e. practiced by Zhang et al. 2016) or vice-versa. Analogous to kinematic viscosity, the thermal diffusivity is an output property based on the given eDPD parameters. In order to measure the thermal diffusivity (α) of eDPD fluids, various methods have been proposed in the literature. For example, Willemsen et al. (2000) have mapped the eDPD solutions to the theoretical solutions of an unsteady heat conduction problem in a 1D semi-infinite region. Recently, Li et al. (2014) have proposed a new way to measure α of eDPD fluid by using the so-called Periodic Heat Conduction (PHC) problem. By referring to the schematic diagram (Figure 4) and applying the external heat source Q_{ext}, α at a particular temperature T_o can be determined by:

$$\alpha(T_o) = \frac{Q_{ext}\left(L_y/2\right)^2}{16C_v}\left(\frac{1}{T_{\max} - T_o} - \frac{1}{T_{\min} - T_o}\right). \tag{21}$$

In the current work, Q_{ext} is 100 (following Li et al. 2014), L_x and L_y are 20 and 30, respectively, and T_o is the initial temperature of the eDPD particle. T_{max} and T_{min} are the simulated maximum and minimum temperatures in the fluid domain, respectively. Figure 6 shows the temperature variation in the y-direction. Accordingly, $T < T_o$ as $y < 15$ because heat is removed from the fluid system. The temperature amplitude correlates with the thermal diffusivity in accordance with Equation (21).

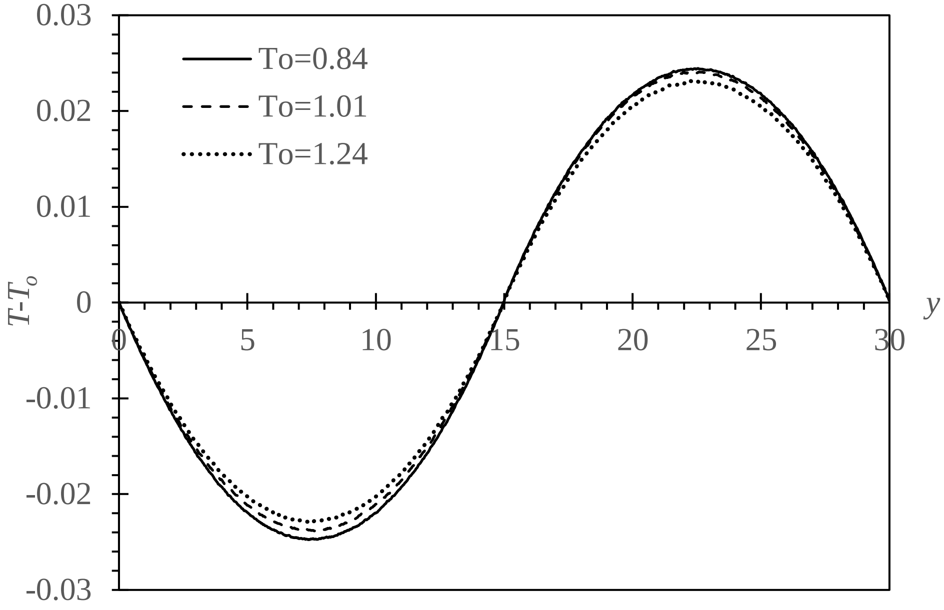
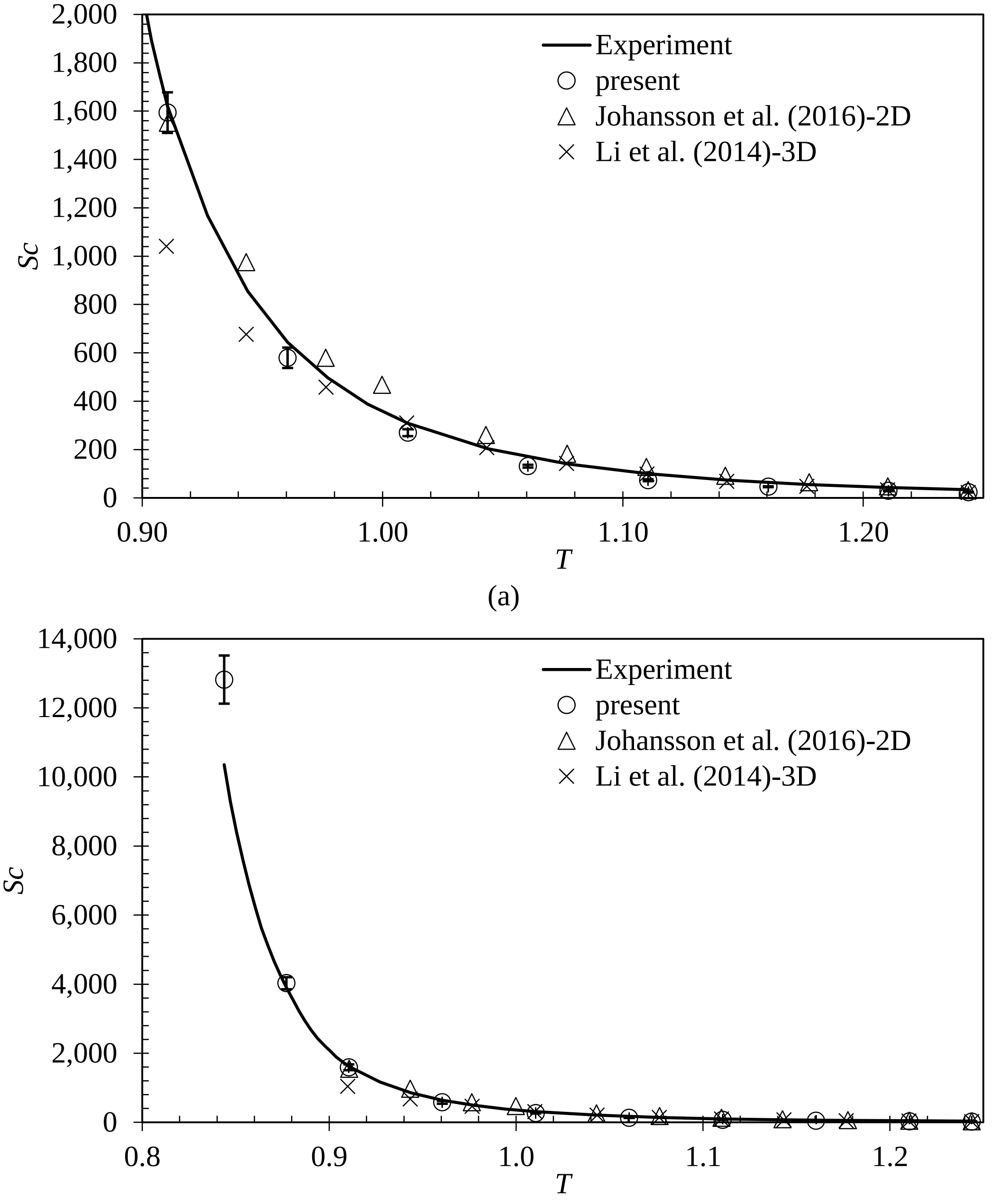


Figure 6: The temperature profiles of the Periodic Heat Conduction (PHC) problems for three temperatures (T_o=0.84, T_o=1.01 and T_o=1.24).

3.3 The Schmidt and Prandtl numbers

By using the measured values of D (Equation (17)), ν (Equation (18)) and α (Equation (21)), the Schmidt numbers ($Sc = \nu/D$) and the Prandtl numbers ($Pr = \nu/\alpha$) of the eDPD fluid at different temperatures can be calculated. Figure 7(a) shows the Sc values obtained by using the current approach. The values obtained by Li et al. (2014) and Johansson et al. (2016) are overlaid on the same figure as well for comparison purpose. As seen, amongst the three approaches, our predictions come closer to the measured values in general, especially for those in low temperature regime ($0.91<T<1.00$). We have repeated the numerical measurement of Sc for 5 times and found that the standard deviation of Sc is fluctuating between 3%-6% of the averaged Sc value. On the other hand, at T=0.84 (see Figure 7(b)), it is noticed that the difference between the predicted and experimental values of Sc is quite discernible. At T=0.84, our estimated s value is ~0.0068 (see Table 1). 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 has prompted Fan et al. (2006) to choose a higher value of s (i.e. s=0.5) in their flow computation. If one intends to closer reproduce the Schmidt number of supercooled water at T=0.84, a simple solution is to increase the values of r_c, γ or ρ in order to get rid of the small s value at T=0.84 at the expense of higher computational cost. This strategy, however, is currently not implemented in this work.

Figure 8 compares the simulated Pr with the experimental data. In general, the agreement is quite promising except at T=0.84 which may be attributed to the reason stated above (s~0). The average error is ~ 6% over the entire temperature range. Here, the standard deviation is generally < 1% of the averaged Pr at each temperature. The standard deviation of Pr at T=0.84 is the highest.



(a)

(b)

Figure 7: Comparison of Schmidt numbers of water for (a) $0.91<T<1.24$ and (b) $0.84<T<1.24$. For the present eDPD simulations, the error bars are obtained by repeating the simulations 5 times. Also, r_c = 1.96, γ = 8.0, ρ=4.

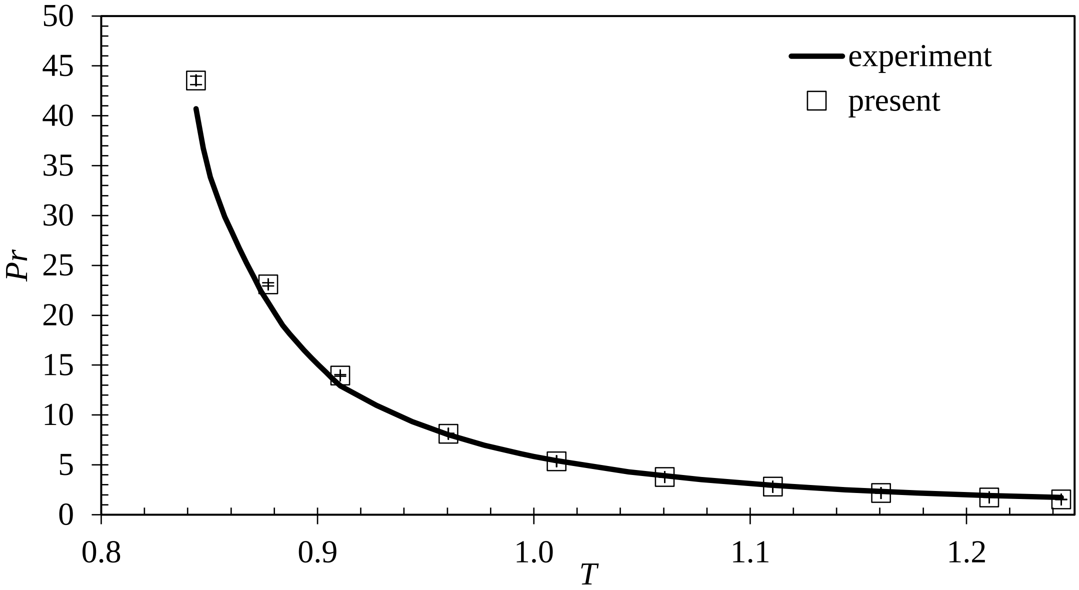


Figure 8: Comparison of Prandtl numbers of water at different temperatures. The error bars of the present eDPD simulations are obtained by repeating the simulations 5 times. Also, r_c = 1.96, γ = 8.0, ρ=4.

By using the methodology stated above, it is out of our curiosity to determine $r_{c,crit}$ for the eDPD parameters used by Johansson et al. (2016). By substituting s=0, γ = 4.5, ρ = 4, T = 0.91(i.e. minimum temperature in the temperature range considered by Johansson et al. (2016)) which correspond to Sc ~1620, we obtain $r_{c,crit}$ ~ 1.7578, which is smaller than the r_c value (=1.81) suggested by Johansson et al. (2016). Therefore, we have chosen r_c =1.76 in our computation and the time step size Δt is increased to 0.01 due to the smaller γ used (Δt similar to that of Johansson et al. 2016). Accordingly, the corresponding temperature-dependent s function is recomputed based on the procedures outlined above as $s^*(T) = 2.5548\, T^3 - 11.443\, T^2 + 20.016\, T - 10.665$ (R^2~1). Figure 9 compares the simulated Schmidt numbers and it is interesting to note that our simulated Schmidt numbers are coming closer to the experimental data even though the chosen r_c is smaller than that suggested by Johansson et al. (2016).

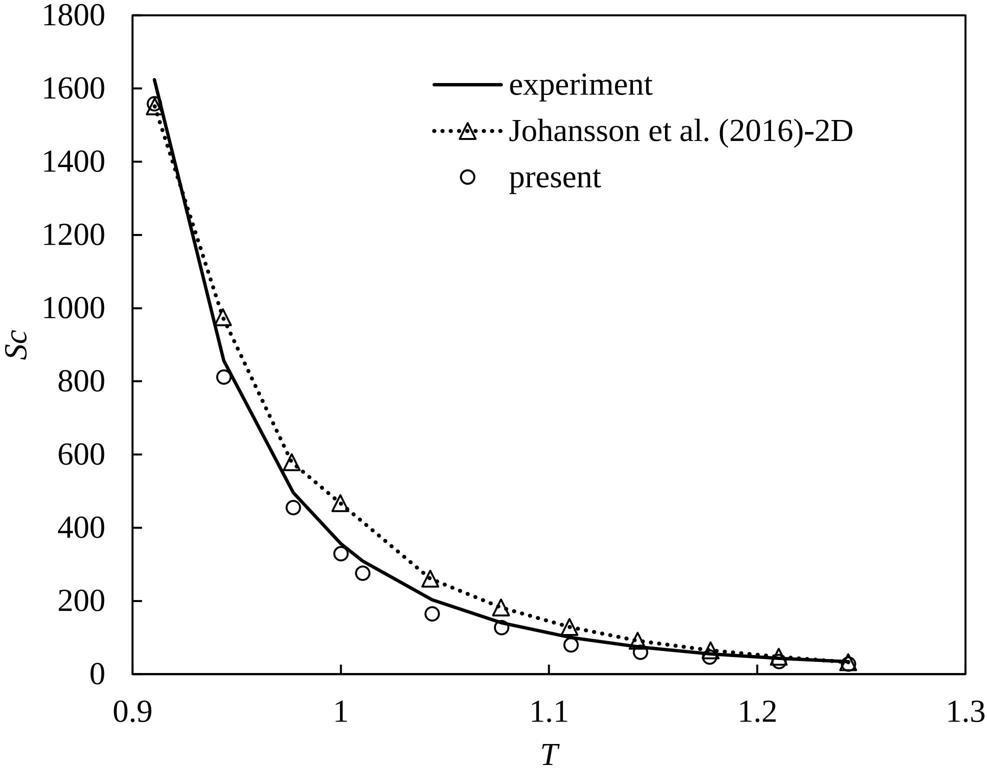


Figure 9: Comparison of Schmidt numbers of water at different temperatures ($0.91 < T < 1.24$). The cut-off radius r_c used for this case ($r_c = 1.76$) is smaller than that of Johansson et al. (2016), i.e. $r_c = 1.81$. $s^*(T) = 2.5548\,T^3 - 11.443\,T^2 + 20.016\,T - 10.665$ ($R^2 \sim 1$) is used in the present computation. $\gamma = 4.5$ and $\rho = 4$.

4.0 Solidification problem

4.1 1D solidification in a semi-infinite plate

Having determined the eDPD parameters from Section 3.0, i.e. $r_c = 1.96$, $\gamma = 8.0$, $\rho = 4$ and $\Delta t = 0.007$, we intend to investigate the 1D unsteady solidification problem in a semi-infinite plate which was previously studied by Johansson et al. (2016) using eDPD. Following their geometric setup, a 2D rectangular domain of size 12x100 is built as shown in Figure 10. The initial temperatures of the eDPD particles are prescribed as $T_o = 0.95$. At $t = 0$, the temperature of the bottom wall is reduced to $T_b = 0.85$ and the liquid particles are allowed to freeze thereafter. Following the approach of Willemsen et al. (2002), the equations of motion are solved only on non-solid particles (i.e. eDPD particles are frozen once they become solid). The time evolution of the solid-liquid interface is then examined. In order to mimic the semi-infinite behavior of the problem, the height of the channel is 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 prevail at the left and right boundaries.

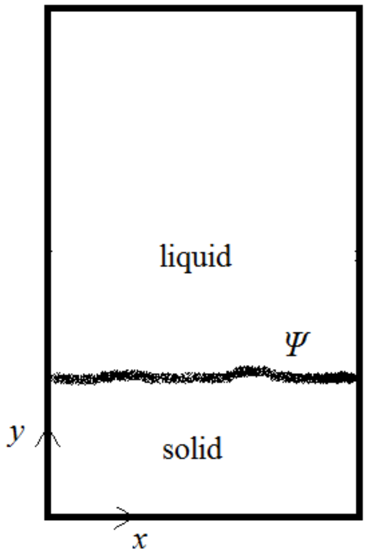


Figure 10: Schematic diagram of the 1D solidification problem. The solid and liquid regions are separated by the solid-liquid interface *Ψ*. Total lengths in *x*-direction (L_x) and *y*-direction (L_y) are 12 and 100, respectively. Periodic boundary conditions are applied on the left and right boundaries. Top and bottom boundaries are treated as walls. The temperatures at the top and bottom walls are 0.95 and 0.85, respectively. The initial temperature is T=0.95.

An analytical solution for this problem is available (Lunardini 1988), whereby the temperature at the solid and liquid regions can be expressed as:

$$T_S(y) = T_b + \frac{T_f - T_b}{erf(\lambda)} erf\left(\frac{y}{2\sqrt{\alpha_S t}}\right) \tag{22}$$

$$T_L(y) = T_t - \frac{T_t - T_f}{erfc(\lambda\sqrt{\alpha_{SL}})} erfc\left(\frac{y}{2\sqrt{\alpha_L t}}\right) \tag{23}$$

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 $(.)_{SL}$ denotes the ratio $(.)_S/(.)_L$. In general, the constant λ determines the speed of the solid-liquid interface *Ψ* as:

$$\Psi = 2\lambda\sqrt{\alpha_S t}\ . \tag{24}$$

As shown by Lunardini (1988), the parameter λ can be solved numerically (iterative method) from the following implicit equation:

$$\frac{e^{-\lambda^2}}{erf(\lambda)} - \frac{k_{LS}\sqrt{\alpha_{SL}}\left(T_t - T_f\right)e^{-\alpha_{SL}\lambda^2}}{\left(T_f - T_b\right)erfc(\lambda\sqrt{\alpha_{SL}})} = \frac{L\lambda\sqrt{\pi}}{C_v\left(T_f - T_b\right)}, \tag{25}$$

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 = 334000J/kg, the latent heat L in DPD unit can be computed by performing the scaling operation as below:

$$\begin{aligned} L &= \frac{L^R}{k_B^R T_*^R / m_*^R} \\ &= \frac{\rho^R l_*^R L^R}{\rho k_B^R T_*^R} \end{aligned} . \tag{26}$$

By substituting the latent heat of water (L^R = 334000J/kg), the density of water ρ^R = 1000kg/m^3, the freezing temperature T_f^R =273.15K and the Boltzmann constant k_B^R =1.3806 x 10^{-23} J/K, the latent heat in DPD unit can be computed from Equation (26) as L=2.68 x 10^4. This value is similar to that obtained by Johansson et al. (2016).

The simulation has been executed until t = 280. In order to compare with the analytical solution, we have 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 is prescribed as 1.1. The mesoscopic heat friction can then be calculated accordingly from Equation (20). Besides that, we intend to examine the speed of solid-liquid interface for different thermal diffusivity ratios as well, i.e. α_{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 are assumed to be temperature-independent in the current work. Therefore, the case of α_{SL} = 4.08 is analogous to the case where $k^R_{T=T_b(ice)} / k^R_{T=T_t(water)} = 4.08$.

In order to study the solidification problem, we have made a slight modification on the state equation initially proposed by Willemsen et al. (2000) in their enthalpy method (via eDPD) for the melting problem. Here, the temperature of particle i is updated as:

$$T_i = \begin{cases} (e_i + L)/C_v & e_i < C_v T_f - L \\ T_f & C_v T_f - L \le e_i \le C_v T_f \\ e_i / C_v & e_i > C_v T_f \end{cases} . \tag{27}$$

Generally, the state equation is used to describe the particle temperature in three distinct fluid regions, i.e. solid region: $e_i < C_v T_f - L$, transition region: $C_v T_f - L \le e_i \le C_v T_f$ and liquid region: $e_i > C_v T_f$. According to the method outlined very recently by Johansson et al. (2016), the states of the particles within the transition region are remained, i.e. liquid/solid particles are retained in liquid/solid state when transition occurs.

We have attempted the approach suggested by Johansson et al. (2016) and the results are shown in Figure 11(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 is determined via identifying the locations of bins with temperature $T_{bin} = T_f$. The mean positions of these bins are then treated as the displacement of Ψ. Each cell is divided into 5 bins in the y-direction, as shown in Figure 12. As observed from Figure 11(a), when α_{SL} = 1.0 and 2.0, the eDPD results are quite close to the analytical solutions. Johansson et al. (2016) have 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 x 10^4) must be subtracted before freezing takes place. However, such behavior is not observed in our current computation, and freezing starts almost instantaneously as suggested by the theoretical solutions. Meanwhile, it is noticed that as α_{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.

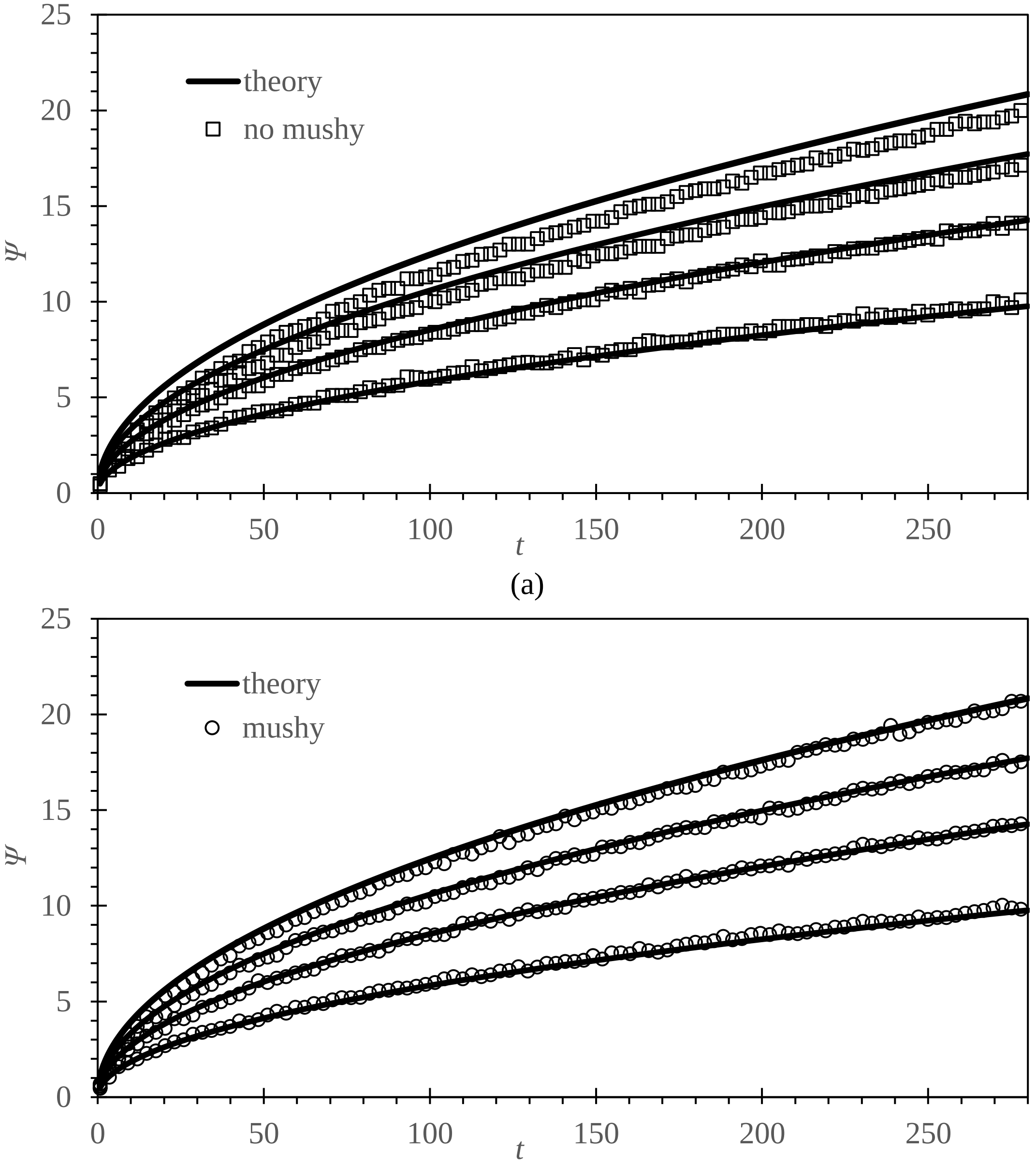


Figure 11: Time evolutions of solid-liquid interface Ψ for cases (a) without mushy particle and (b) with mushy particles at different thermal diffusivity ratios (α_{SL}), i.e. 1.0, 2.0, 3.0 and 4.08. As α_{SL} increases, the speed of Ψ increases.

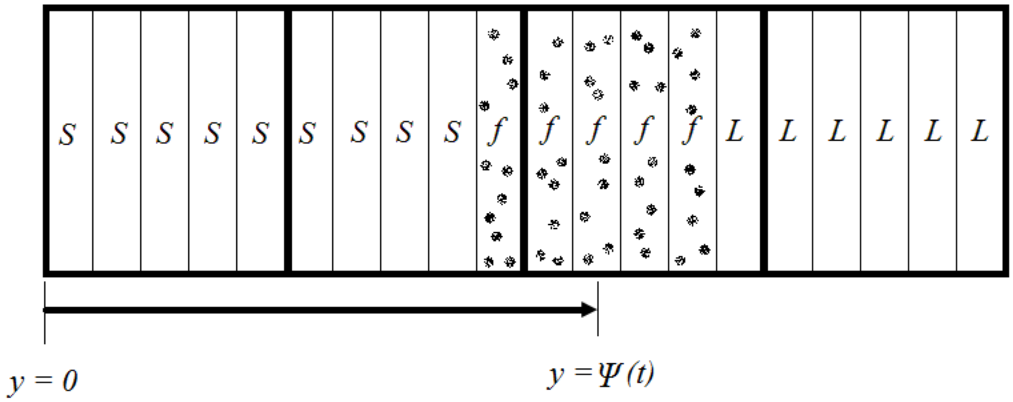


Figure 12: The displacement measurement of solid-liquid interface $\Psi(t)$. S, L and f denote respectively the solid phase, liquid phase and transition phase (mushy zone). Each cell (bounded by thick solid line) is divided into 5 bins in the y-direction. The temperature in the transition zone is T_f.

In fact, it has been well appreciated that 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, the 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, the particle is partially solid and partially liquid ("mushy"). Figure 13 shows the thickness of the mushy layer (total size of bins with $T = T_f$) as time progresses. 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 cut-off radius r_c. Owing to this, we argue that the result might 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 particle i, namely φ_i in the transition zone as:

$$\varphi_i = \frac{C_v T_f - e_i}{L} \tag{28}$$

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

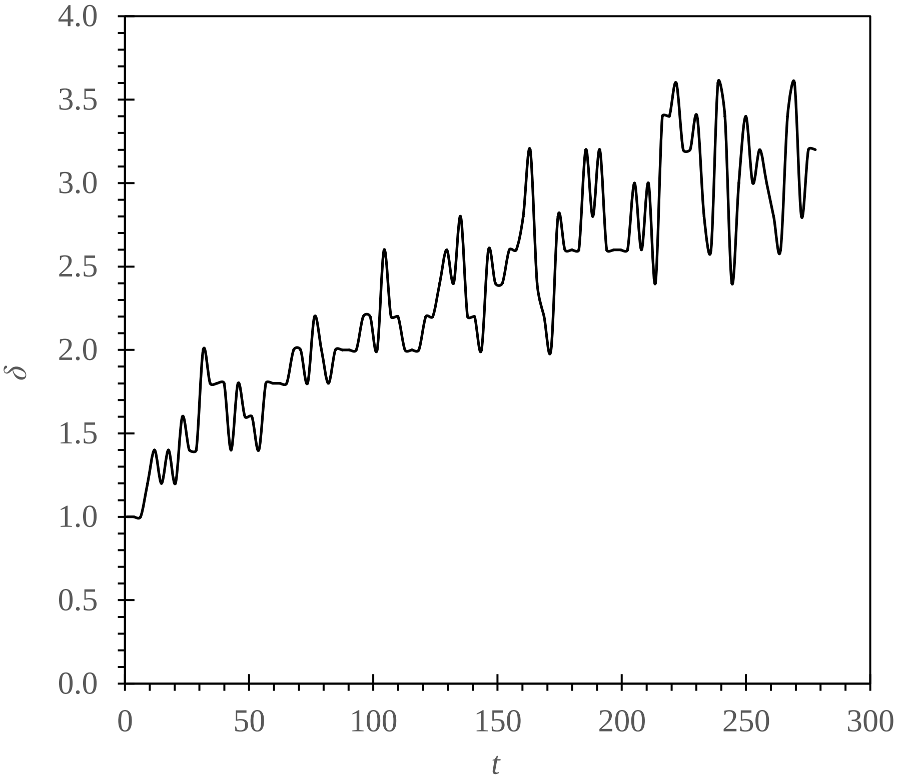
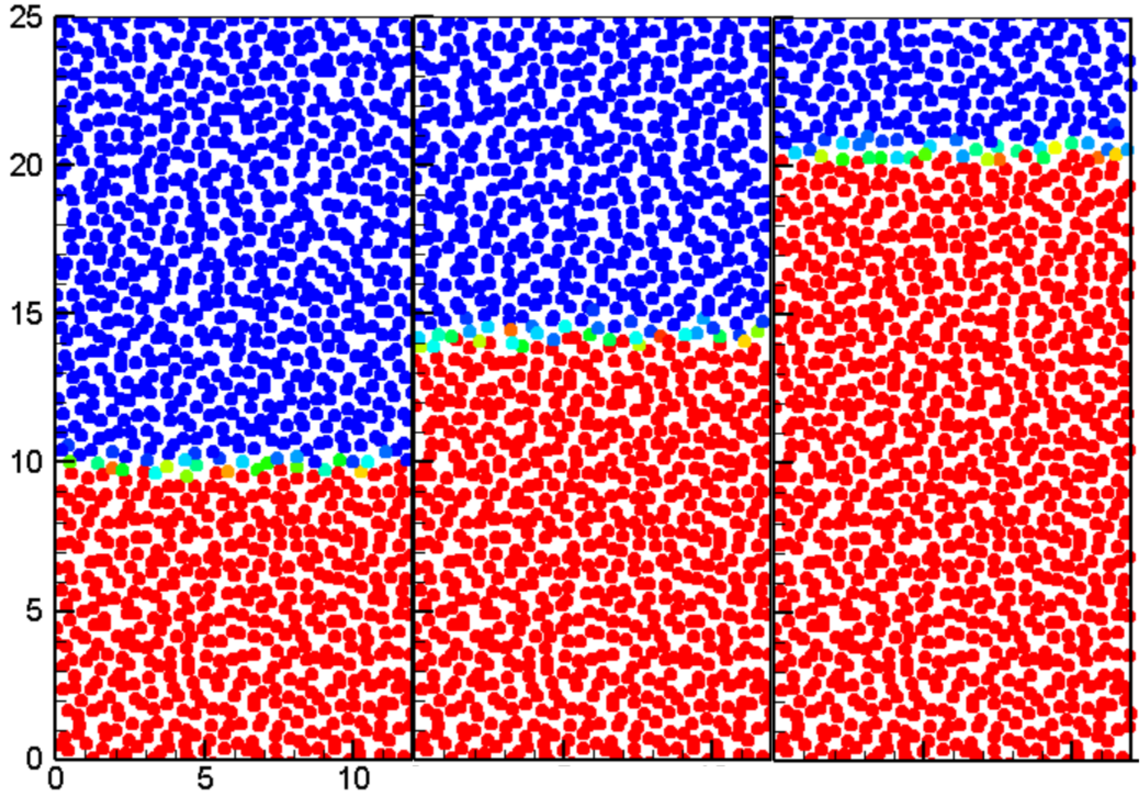


Figure 13: The growth of mushy layer thickness as time elapses. α_{SL} = 1.0.

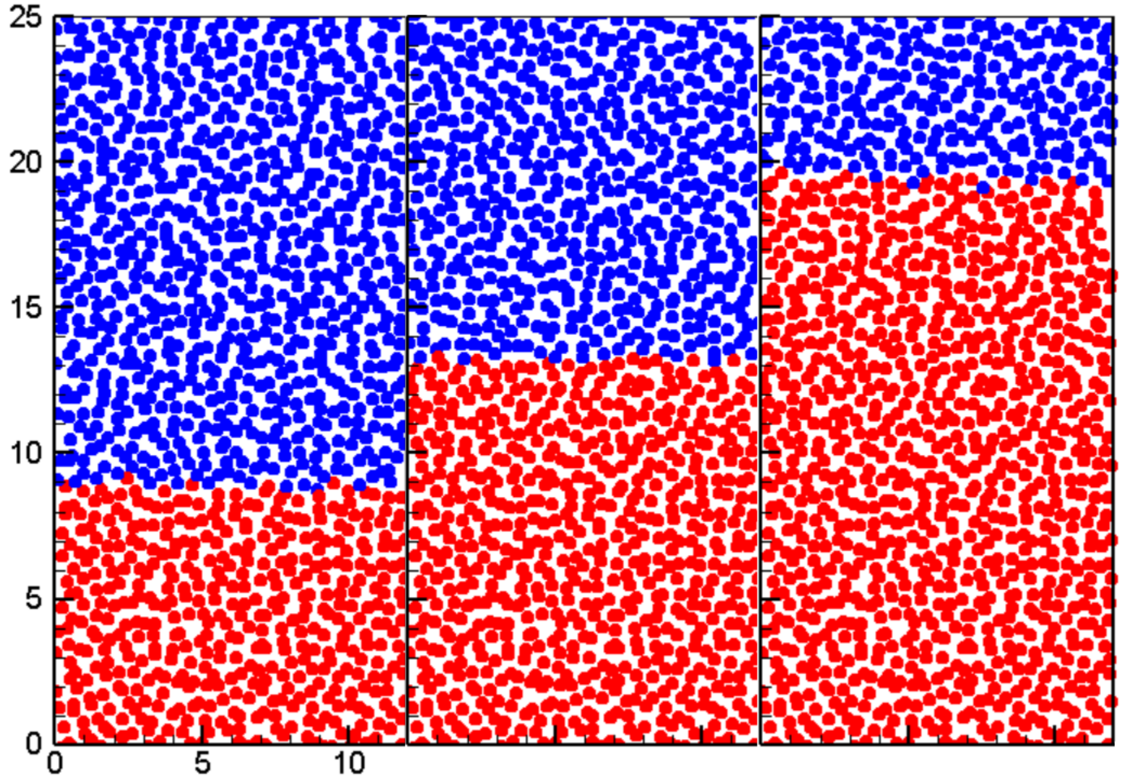
Now, the underlying challenge is on how to determine the thermal diffusivity of mushy particle. According to Alexiades and Solomon (1993), the effective thermal diffusivity of a mushy particle relies heavily on the structure of the solid-liquid interface. Some formulations have been proposed in the numerical framework of finite-difference schemes. However, in the context of particle method such as eDPD, it is unclear on how to define such structure. Accordingly, we have linearly interpolated the thermal diffusivity of mushy particle from α_S and α_L based on the computed solid fraction:

$$\alpha_i = \varphi_i \alpha_S + (1 - \varphi_i)\alpha_L \, . \tag{29}$$

A similar idea has been pursued in other particle method such as the Finite Volume Particle (FVP) method (Mahmudah et al. 2011). Figure 11(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 Figure 14 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.



(a)

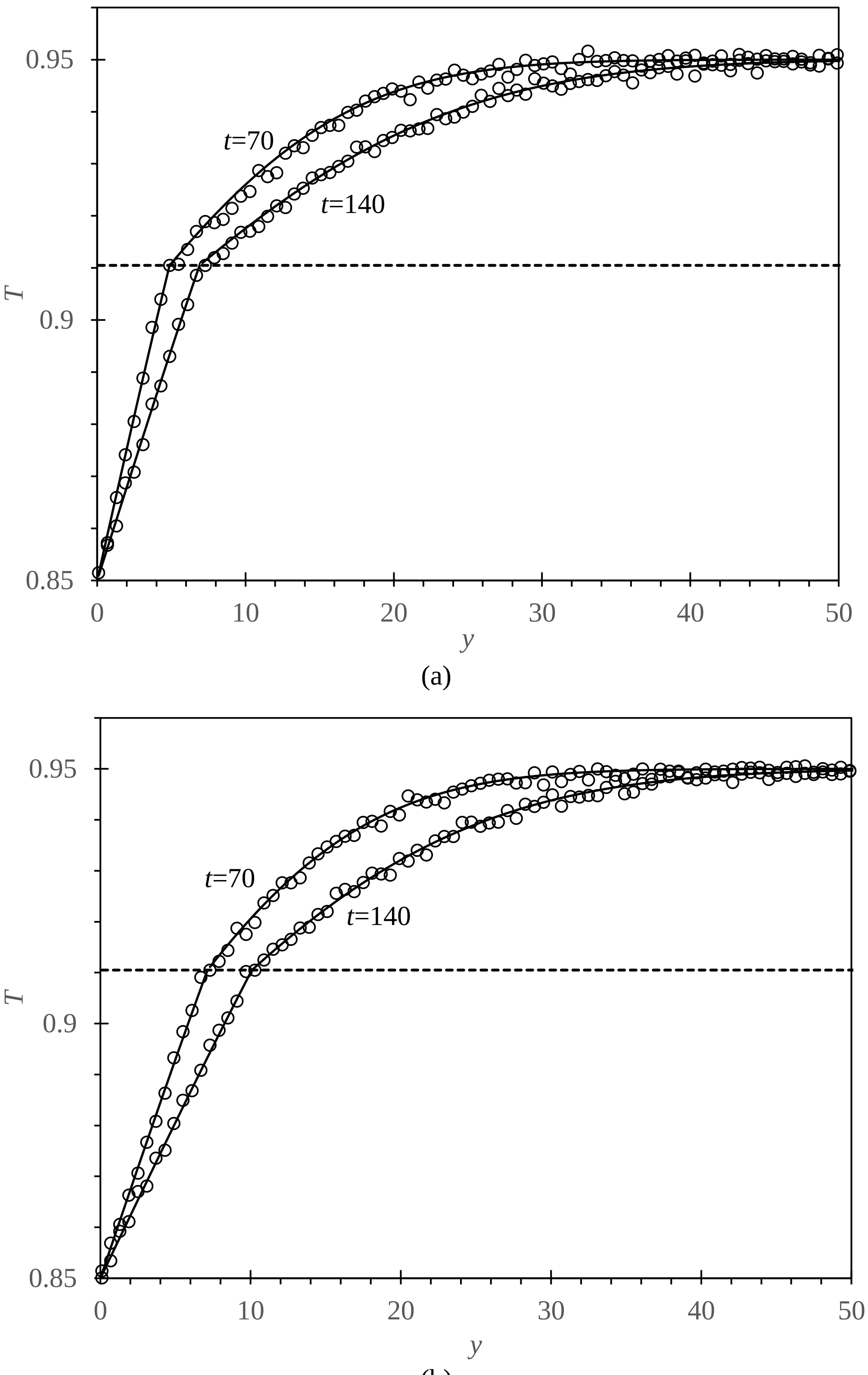


(b)

Figure 14: Instantaneous positions of the eDPD particles at *t*=70 (left), *t*=140 (middle) and *t*=280 (right) coloured by solid fraction φ_i for cases (a) with mushy and (b) without mushy particles. Red: solid phase. Blue: liquid phase. The solid-liquid interface is moving in the $+y$ direction.

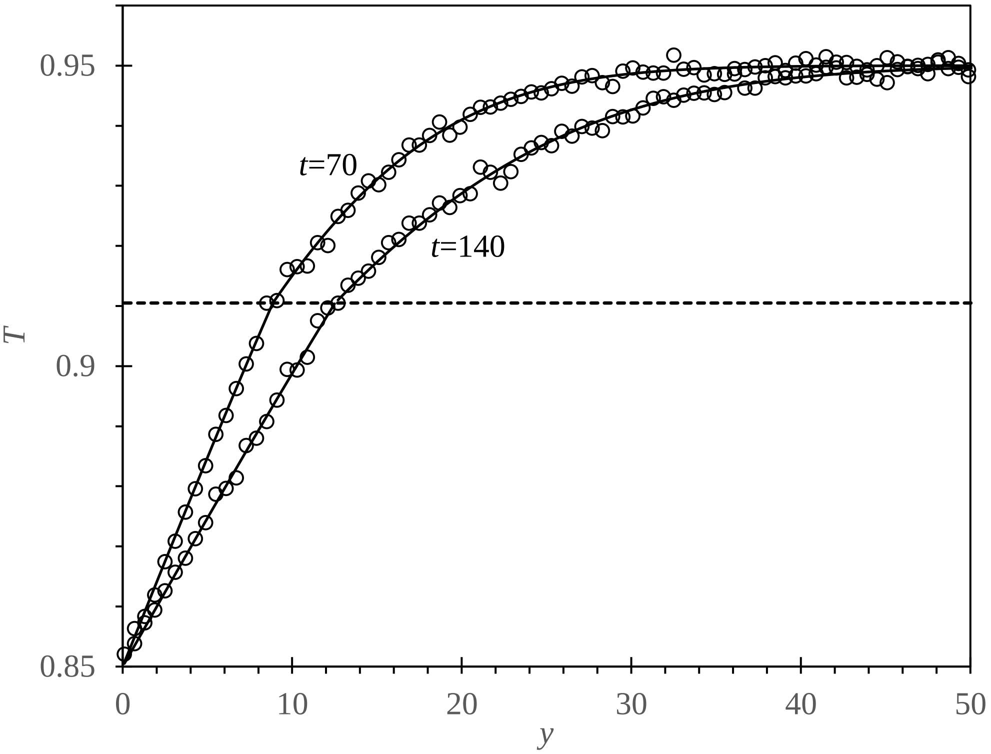
We have further compared our computed temperatures at different time levels with the theoretical solutions. Despite some statistical fluctuations, the predictions (Figure 15) show good agreement with the theoretical solutions in general. We have noticed that the near-wall temperature profile reported by Johansson et al. (2016) is not reproduced correctly as their predicted temperature experiences a sharp increase near the wall (c.f. Figure 3 in Johansson et al. 2016). Willemsen et al. (2000) have addressed this problem by reflecting the near-wall fluid particles about the wall axis. The new reflected particles are then assigned a temperature value in which the mean temperature of the original and the new particles equals to 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 particles. In the current work, we have generated an extra layer of wall particles located within the wall region and their positions are fixed during the course of the computation. In order to obtain their temperatures, we reflect the wall particles about the wall axis to form a new internal “imaginary” particle. Note that the positions of these imaginary particles are non-varying because the positions of wall particles are fixed. The temperature value of wall particle is then assigned accordingly similar to that proposed by Willemsen et al. (2000) by assuring that mean temperature of the wall particle and the imaginary particle equals to the prescribed wall temperature. Here, the temperature of imaginary particle is interpolated from the interior eDPD particles by using the Moving Least Square (MLS) method (Ng et al. 2016). The schematic diagram of our implementation of boundary condition is shown in Figure 16. As reported in Figure 15, the predicted near-wall temperature is reproduced correctly for different α_{SL}, which is consistent to the boundary condition ($T_b = 0.85$) imposed.

Also, we have performed averaging on the unsteady temperature profiles for case $\alpha_{SL} = 4.08$ based on 15 simulations generated from different random seeds as shown in Figure 17 in order to eliminate the statistical noises. 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. (2016). Meanwhile, it is appealing to note that the numerical results obtained by using mushy particle come closer to the analytical solution.

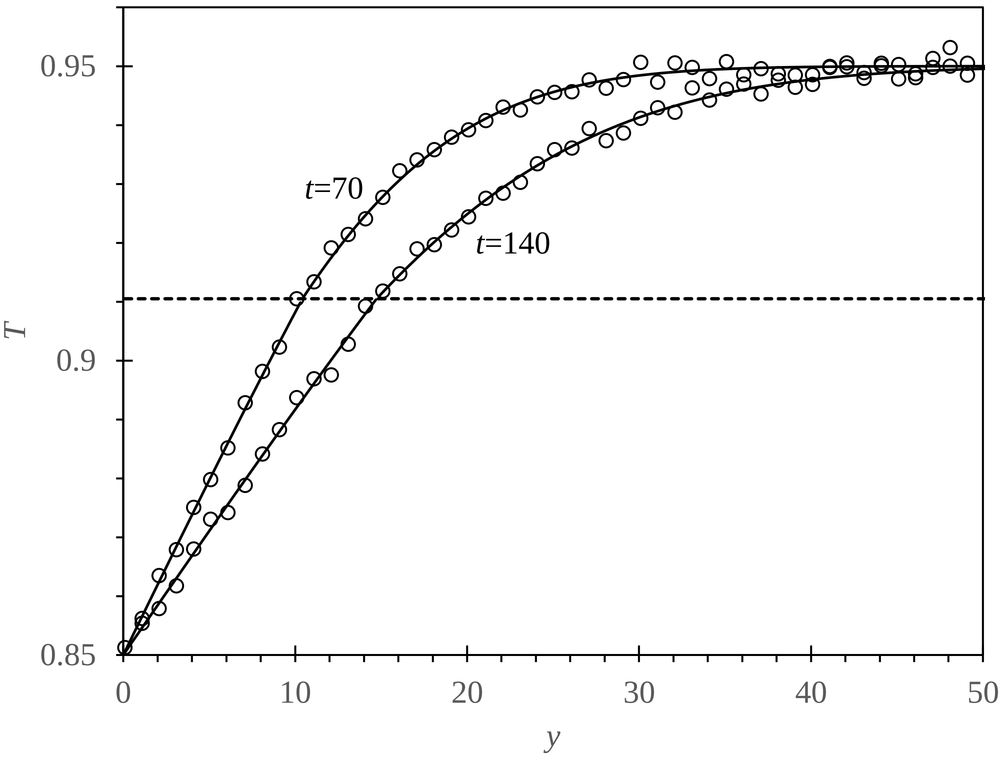


(a)

(b)


0.95
0.9
0.85
T
0
10
20
30
40
50
y
t=70
t=140

(c)


0.95
0.9
0.85
T
0
10
20
30
40
50
y
t=70
t=140

(d)

Figure 15: Temperature distribution along the y-direction for α_{SL} = (a) 1.0, (b) 2.0, (c) 3.0 and (d) 4.08. The dash line indicates the $T = 0.9105$ (freezing temperature). eDPD method with mushy particle is used.

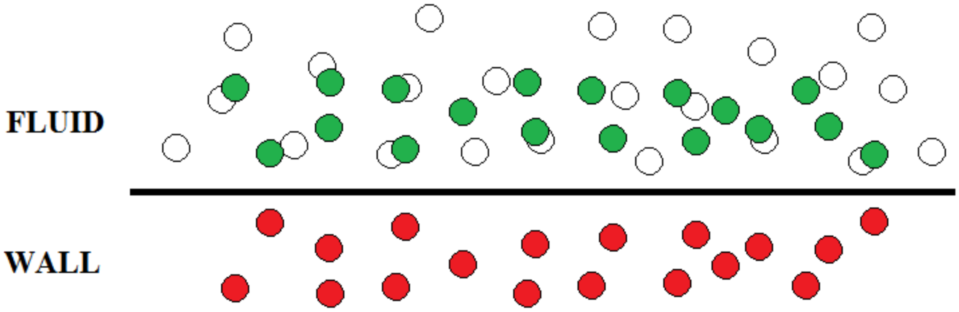


Figure 16: Wall boundary condition. Red: wall particles. Green: Imaginary particles (reflection from wall particles about the wall boundary). Empty circle: interior fluid particle. Temperatures of imaginary particles are interpolated from interior fluid particles.

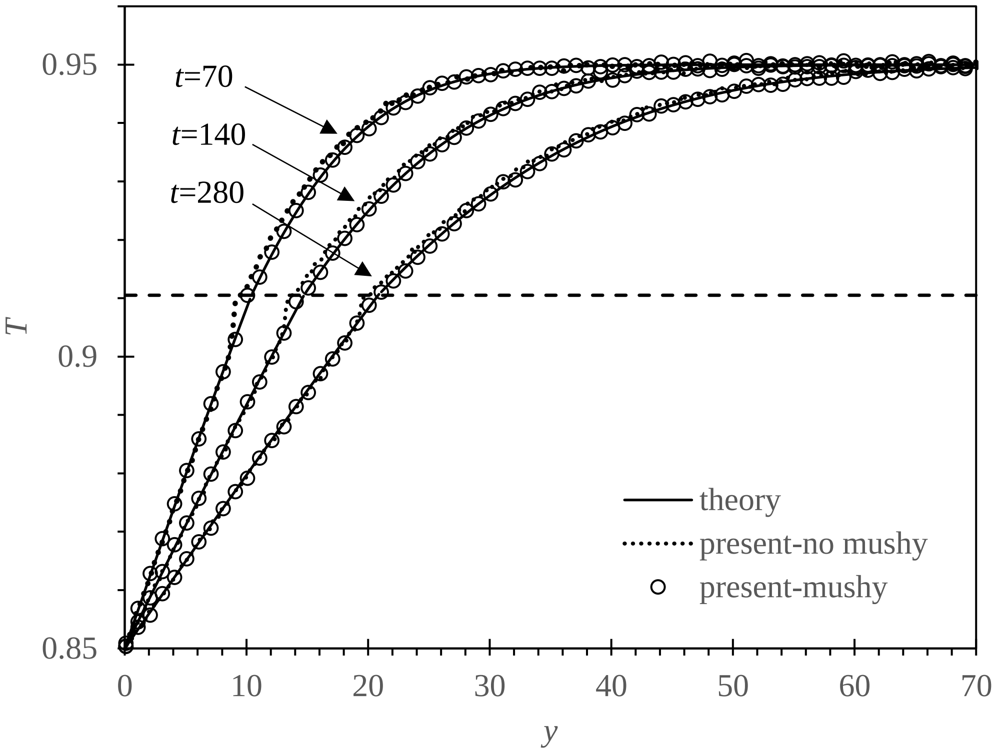


Figure 17: Temperature distributions along the y-direction for α_{SL} = 4.08 at different time levels. The dash line indicates $T = 0.9105$ (freezing temperature). eDPD methods with/without mushy particles are used. The temperatures are averaged by 15 sample data.

4.2 2D solidification problem

Next, we intend to simulate the 2D solidification problem in a square cavity. So far, according to the best of our knowledge, this problem has not been addressed by the eDPD method. Figure 18 shows the schematic diagram of the problem.

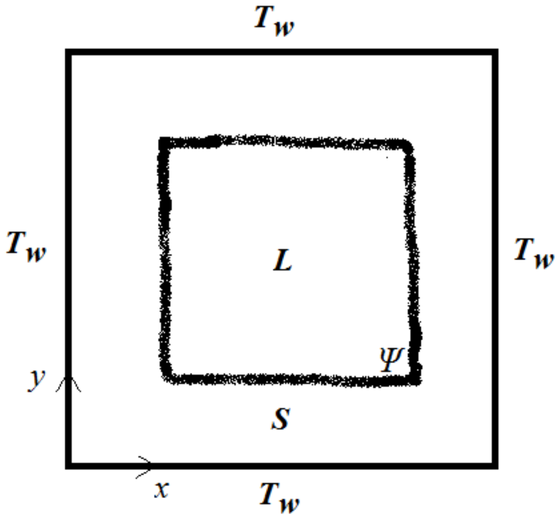


Figure 18: Schematic diagram of the 2D solidification problem. The solid (*S*) and liquid (*L*) regions are separated by the solid-liquid interface Ψ. Both lengths in the *x*-direction (L_x) and the *y*-direction (L_y) are 30. The wall temperatures are T_w=0.8509, which are lower than the initial fluid temperature T_o=1.0296.

Rathjen and Jiji (1971) have solved the 2D solidification problem on a semi-infinite domain analytically. They have defined the Stefan number and the dimensionless initial temperature T_o^* as

$$St = \frac{C_v\left(T_f - T_W\right)}{L}, \tag{30}$$

and

$$T_o^* = \frac{T_o - T_f}{T_f - T_W}, \tag{31}$$

respectively, assuming that the thermal conductivities of both phases are similar. Here, the thermal diffusivity of α = 1.1 is used. 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 is dropped to T_W (below the freezing temperature) to initiate the solidification process. Figure 19 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. $x^* = x/\sqrt{4\alpha t}$ and $y^* = y/\sqrt{4\alpha t}$ of the mushy particles for region: (0<x<15, 0<y<15) at different time levels are compared to the analytical solution in Figure 20. It is interesting to note that the numerical data collapse on the analytical solution for t< 21, indicating that our results compare well with the semi-infinite solution during the initial phase of the solidification process occurred in the current domain. The dimensionless positions of the mushy particles for $t>21$ are shown in Figure 21. Now, as expected, the simulated data no longer follow the theoretical solution due to the finite domain size considered in the current computation. As seen from Figure 19(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 are gaining their sizes and makes the interface's shape closer to a circle as witnessed in Figure 19(d). In fact, the interface is thickening as seen in Figures 21(a-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 Figure 21(c), forming a somewhat “circular-like” transition zone. Finally, this “circular-like” transition region shrinks as seen from Figures 21(c-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 Figures 21(a-f).

The change of the number of mushy particles is reported in Figure 22. Here, we have performed 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 particle decays in a somewhat linear fashion. Generally, the decay rate of mushy particle is smaller than its growth rate.

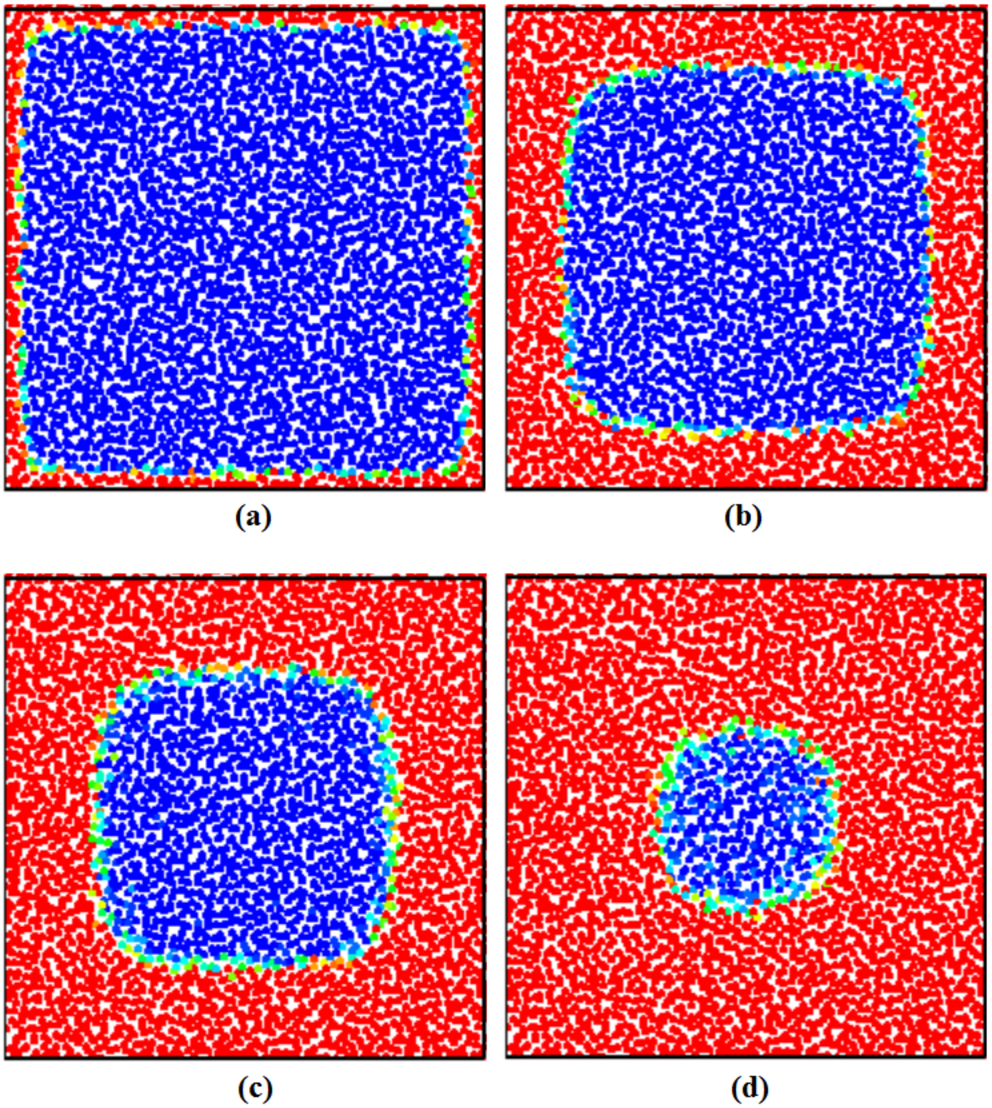


Figure 19: The solidification process in a 2D square domain. The positions of solid-liquid interface at time (a) t=7, (b) t=56, (c) t=105 and (d) t=210 are shown. Solid and liquid particles are marked in red and blue colours, respectively.

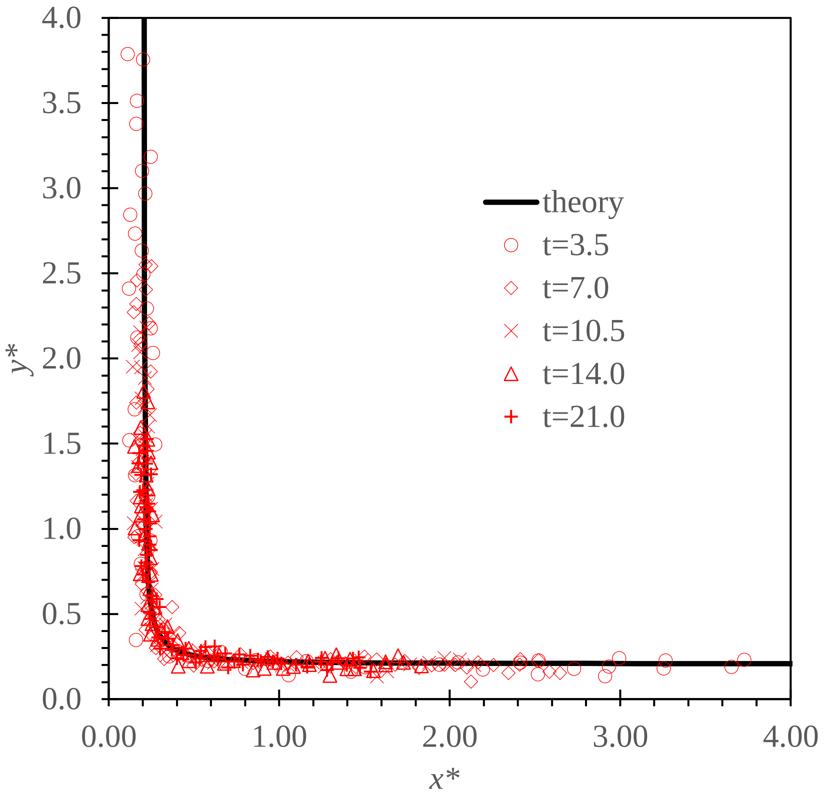
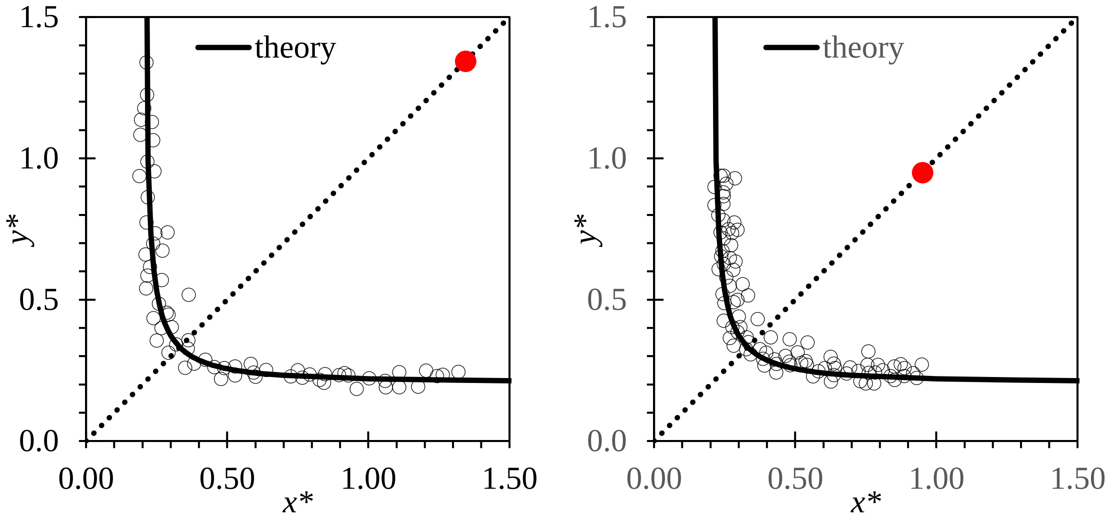


Figure 20: Comparison of the instantaneous dimensionless positions of mushy particles (x^*, y^*) with the analytical solution at different time levels.



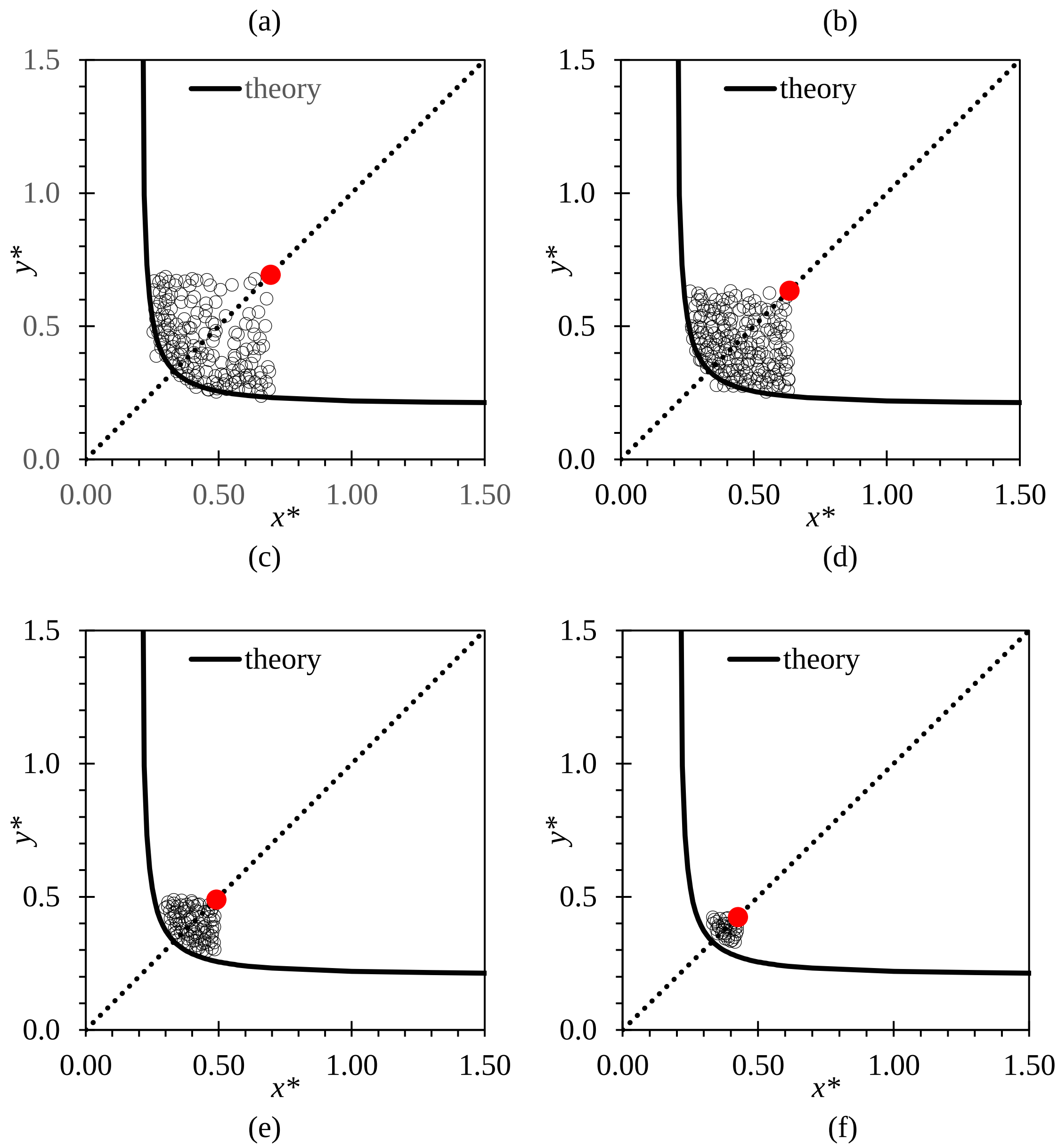


Figure 21: Comparison of instantaneous dimensionless positions of mushy particles (x^*, y^*) with the analytical solution for (a) t=28, (b) t=56 (c) t=105, (d) t=126, (e) t=210, and (f) t=280. The red solid circle signifies the dimensionless position of the square center. For these particular instants, the numerical solutions do not agree with the analytical solutions due to the semi-infinite assumption made in the latter.

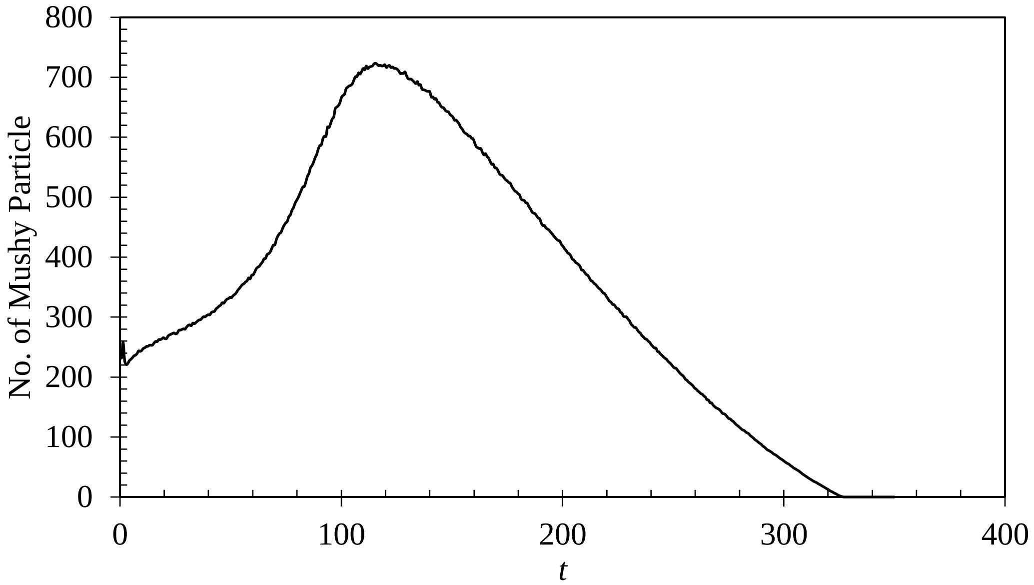


Figure 22: Total number of mushy particles during the 2D solidification process.

4.3 Solidification in a flowing channel

In this Section, we intend to study the solidification behavior of a flowing water in a channel subjected to cooling. Figure 23 shows the schematic diagram of the flow problem. Following the approach of Ge et al. (2015) in their molecular dynamics simulation, the liquid pump and the temperature reset regions of lengths L_P and L_T are placed adjacent to the main solution domain (size L_x x L_y) as shown in Figure 23.

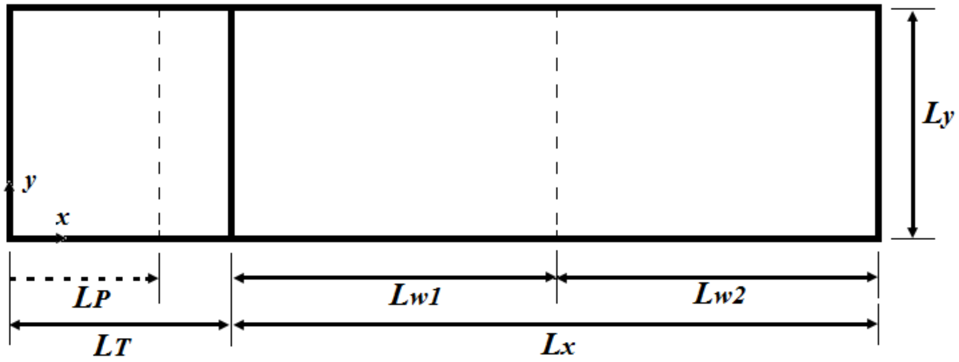


Figure 23: Schematic diagram of the flow channel. The main solution domain is of size L_x x L_y. The size of the computational domain is (L_T + L_x) by L_y. Here, $L_T + L_x = 220$ and $L_y = 20$. L_P, L_T, L_{w1} and L_{w2} are the lengths of the liquid pump region, the temperature reset region, the first wall region and the second wall region, respectively.

Before we study the solidification process in the channel, we wish to validate our implementation of the temperature reset region. Very recently, Zhang et al. (2016) have studied the conjugate heat transfer in a 2D channel. They have placed a temperature reset region of length $L_T = 20$ upstream of the main solution domain of size $L_x = 200$ and $L_y =$

20. Also, they have considered the Poiseuille flow problem by ensuring all the fluid particles in the solution domain are subjected to an external force vector $\vec{F}^{E}$ = <F_{ext}, 0.0>, where F_{ext} = 0.04 (i.e. $L_P = L_T + L_x = 220$). Periodic boundary conditions are prescribed at the left and right boundaries while no-slip conditions prevail at top and bottom boundaries. The temperatures of the eDPD particles in the temperature reset region are fixed at T_r=1.0 before they enter the main solution domain (L_x x L_y). The wall temperatures in temperature reset region are set to T_r as well, while T_{w1} and T_{w2} are prescribed as 1.4. Also, it is important to note that there is no heat exchange between the left and right boundaries to avoid heat from diffusing upstream from the temperature reset region.

In order to reproduce the results of Zhang et al. (2016), we have modified our eDPD parameters to $r_c = 1.0$, $s = 2.0$, $\sigma = 3.0$, $\kappa = 1.26 \times 10^{-4}$ (or $\alpha = 1.3195$, see Equation (20)), similar to those of Zhang et al. (2016). The time step size Δt is set to 0.007 and the simulation is executed for 150000 time steps to obtain a statistically steady result. Time-averaging is then performed for another 20000 time steps.

Figure 24 shows the time-averaged u-velocity in the y-direction of the channel. Based on the simulated u_{max} (=8.08), the theoretical solution $u(y) = u_{\max}\left[1-\left(\frac{y-h}{h}\right)^2\right]$ is used to fit the numerical data. Here, h is the half channel height, i.e. $h = 0.5L_y$. The simulated kinematic viscosity $\nu = F_e h^2 / 2u_{\max}$ is 0.2475, which is ~3% lower than that reported by Zhang et al. (2016): $\upsilon = 0.255$. The discrepancy may be attributed to the difference of wall treatment method, whereby Zhang et al. (2016) have placed the wall particles in the staggered manner and specified a different repulsive force parameter (a_{fs}) between the fluid and the wall particles. Despite its simplicity, this method involves rigorous fine-tuning (both the arrangement of wall particles and a_{fs}) and the optimized parameters are indeed problem-dependent. In the current approach, we place the wall particles in a random manner, which is observed to yield a smaller slip velocity near the wall. Similar observation has been reported by Moeendarbary et al. (2008). Also, we found that the near-wall density fluctuation can be suppressed by introducing randomness while positioning the wall particles. Bounce-back operation is performed when fluid particles penetrate through the walls. Figure 24 shows the time-averaged density plot as well and it is appealing to note that the density fluctuation near the wall is mild.

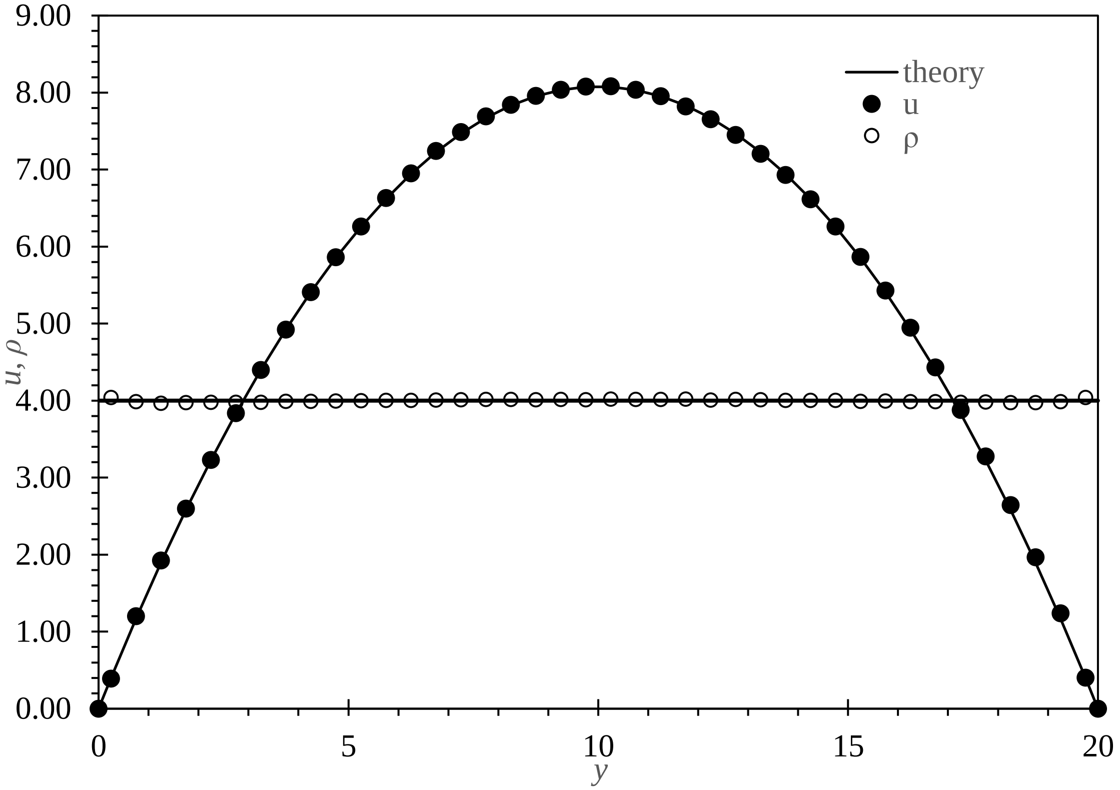


Figure 24: Time-averaged velocity and density profiles in the flow channel.

In order to validate the temperature profile obtained, we have simulated the same flow configuration by using the Finite Volume (FV) method. The flow Reynolds number is the same as that of eDPD, i.e. $Re = u_{max}L_y/\upsilon = 653$. It is unnecessary to consider the temperature reset region in FVM. Here, we have prescribed the parabolic flow profile (obtained from eDPD) at the inlet of the main solution domain of the FV model and fixed the inlet temperature and the wall temperatures (T_{w1} and T_{w2}) to T^R=300K (T = 1.0) and T^R=420K (T = 1.4), respectively. Outflow boundary condition is specified at the flow outlet. Figure 25 compares the temperatures from the current eDPD method and FVM at different flow sections placed throughout the main solution domain. The agreement between both sets of result is promising.

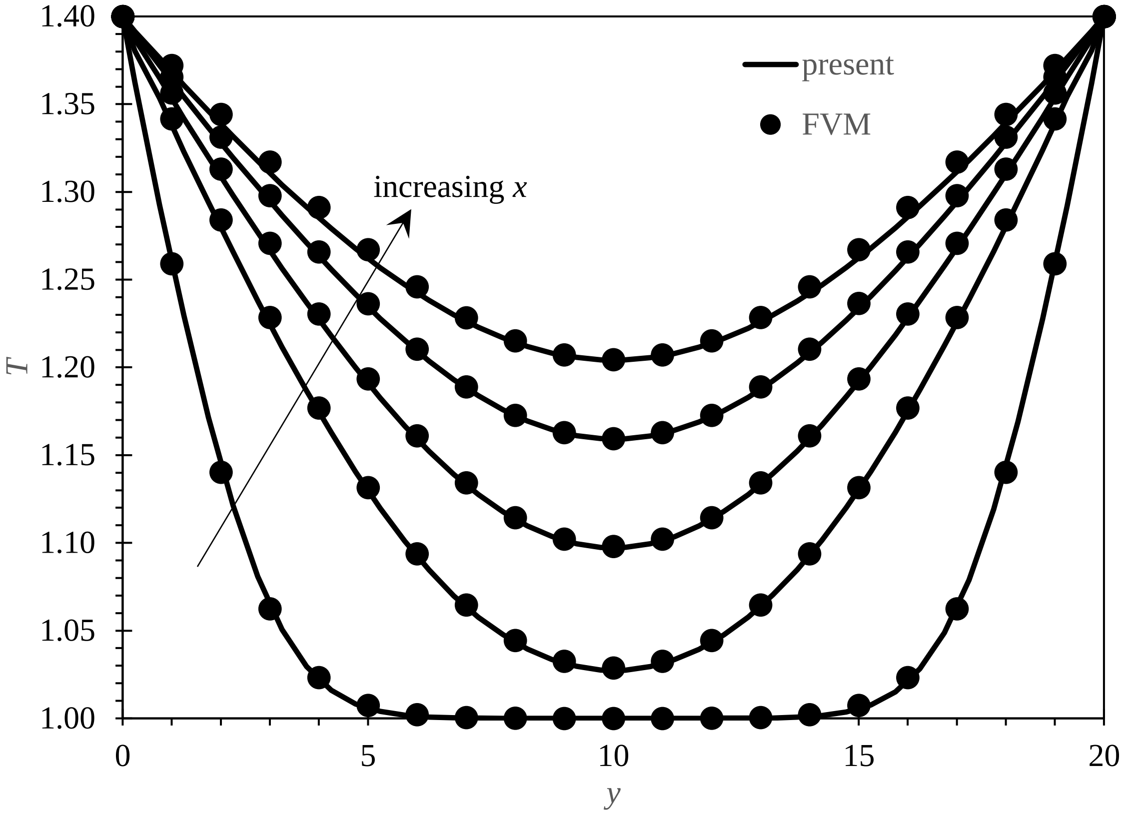
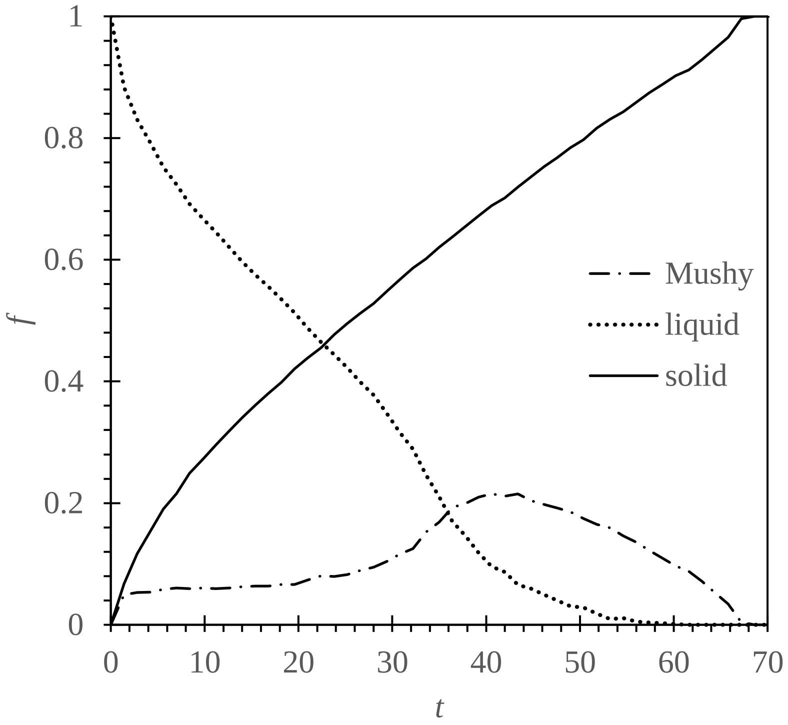


Figure 25: Time-averaged temperature profiles in the flow channel at various cross sections: x= 25, 70, 120, 170 and 215.

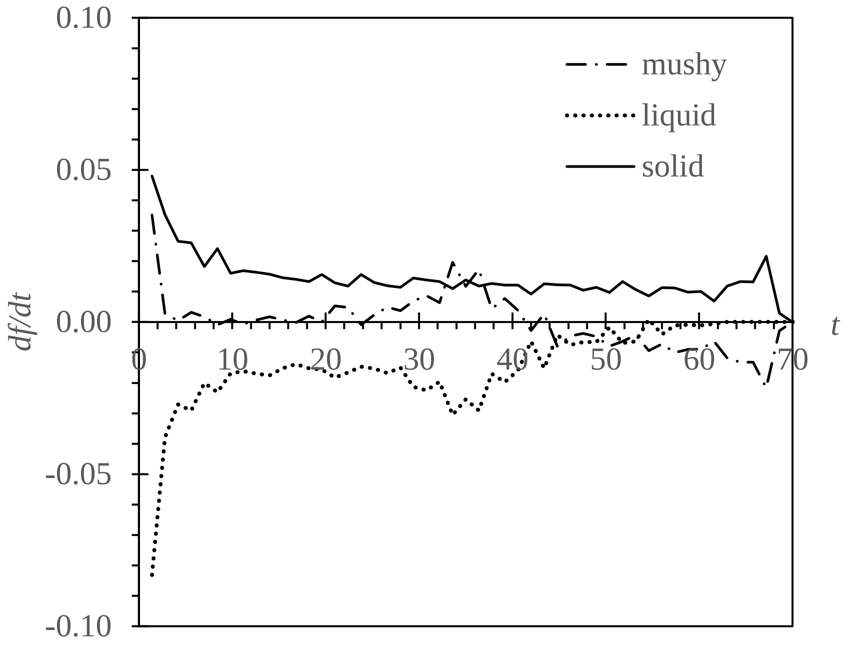
Upon validating the implementation of the temperature reset region in eDPD, the same method is then applied to simulate the solidification process in the flow channel. At this point, it is worth to mention that the thermal conductivity of ice increases as temperature decreases. Accordingly, by following the scaling approach (i.e. ratio of thermal conductivity) proposed by Zhang et al. (2016), we have computed the mesoscopic heat friction of ice particles via: $\kappa_i(T) = \left(k_T^R / k_{T=1.0}^R\right)\kappa_{T=1.0}$, where k_T^R is the experimental data of thermal conductivity measured at temperature T.

Following the approaches of Ge et al. (2015) and Johansson et al. (2016), a liquid pump region is placed upstream of the main solution domain. The length of the liquid pump region L_P is set to 36. Meanwhile, L_T is set to 40. A streamwise external force of F_{ext} = 0.5 is applied on fluid particles in the liquid pump region in order to drive the flow. Firstly, we reset the eDPD parameters to those suggested in Section 4.1 and execute an isothermal flow simulation by setting the temperature of the entire flow domain to T=1.0 (including the wall temperatures). The flow computation is carried out for 170000 time steps (t = 1190) and the velocity is time-averaged after t=700. For this case, u_{max} is ~0.62 and hence Re ~ 1.83. After the flow equilibrium is achieved, the wall temperature T_{w1} (temperature of the first wall segment of length L_{w1} = 100) is reduced to 0.85 and the solidification process is simulated. Here, we are interested to obtain the relationship between the wall temperature T_{w1} and the channel closing time (t_c).

Figure 26(a) shows the fractions of mushy (f_M), liquid (f_L) and solid (f_S) particles within the fluid zone: $70<x<100$ for case $T_{w1} = 0.85$ (intense cooling). The channel is considered to be closed when $f_L = 0$. As seen, f_M is experiencing a slow growth initially ($t<30$). Figure 26(b) reveals that the growth rate of f_M is ~ 0 within $5<t<20$ (see also the illustrations in Figures 27(a-c)). In other words, the decay rate of the liquid particle is almost equivalent to the growth rate of solid particle within this duration. As the solid layers grow from the top and bottom walls and approach each other within $30<t<40$ (Figures 27 (d,e)), more liquid particles are transformed into their mushy state due to the cooling from the opposing solid layer. During this process, one can see a noticeable increase in f_M and it peaks at t~40 (Figure 27 (f)). Accordingly, a higher decay rate of f_L can be seen from Figure 26(b) within $30<t<35$. As more mushy particles are transformed into solid (ice) and the remaining liquid particles become lesser, f_M starts decaying beyond $t=40$ (Figures 27(g-h)). At $t=60$, liquid particles are no longer available within this region, further accelerating the decay rate of f_M. Continuous cooling solidifies the remaining mushy particles and the full solid is formed at t~68 (Figure 27 (i)).

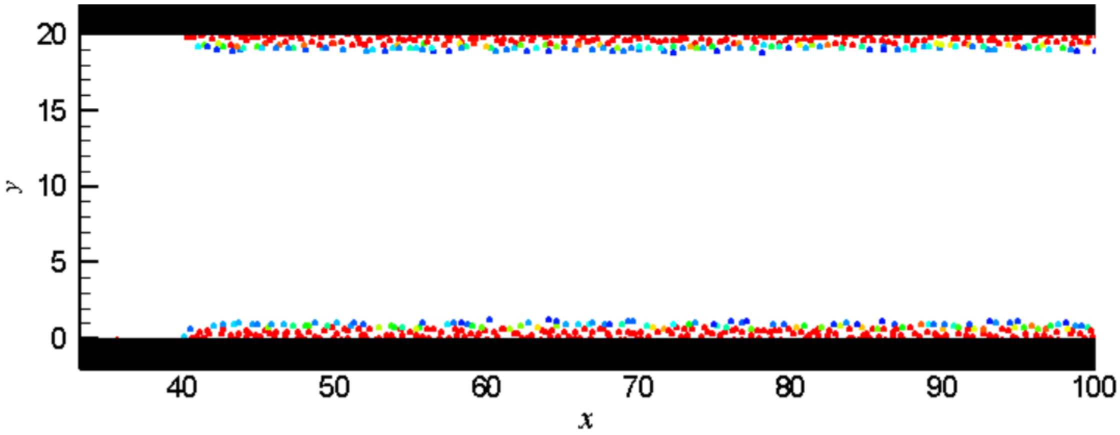


(a)

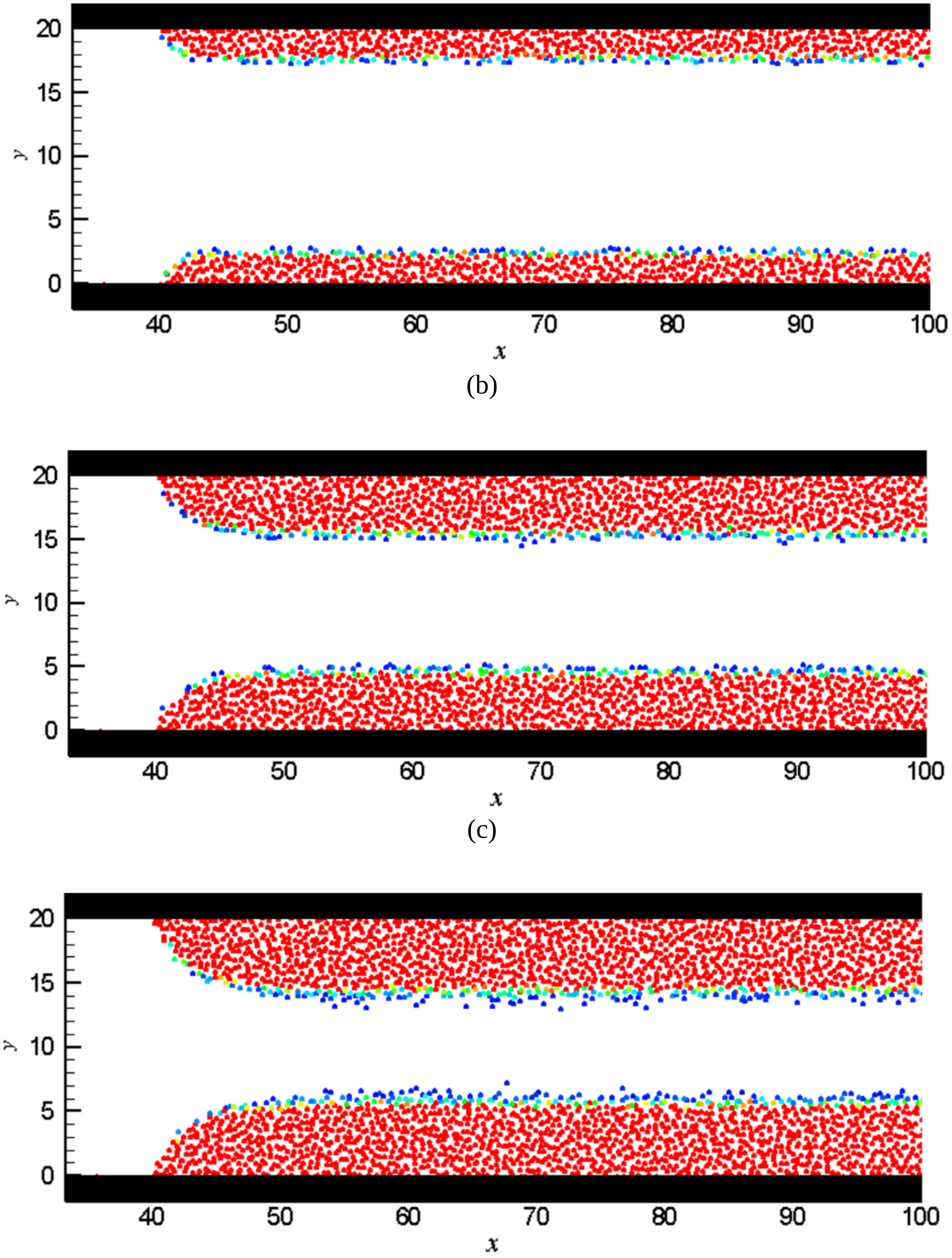


(b)

Figure 26: Time histories of (a) phase factions and (b) rates of change of phase fractions within the region: $70 < x < 100$. $T_{w1} = 0.85$.



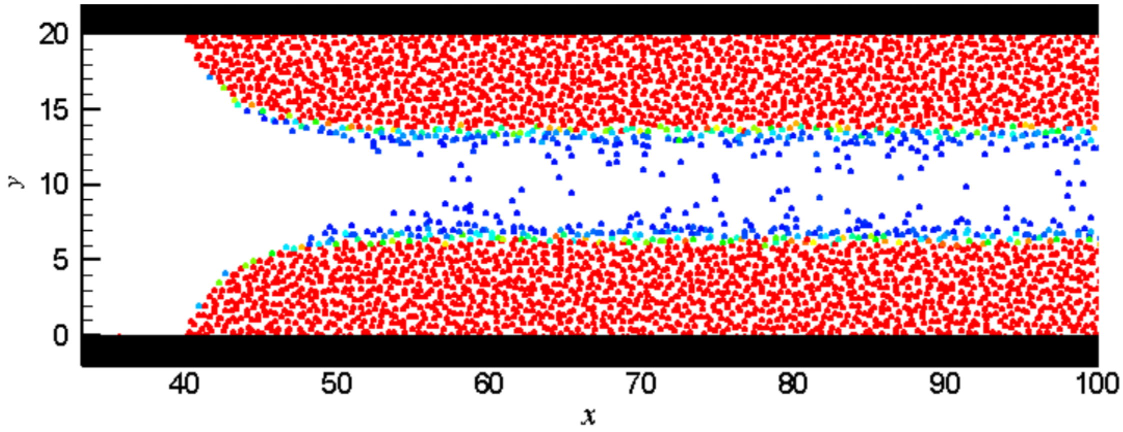
(a)



(b)

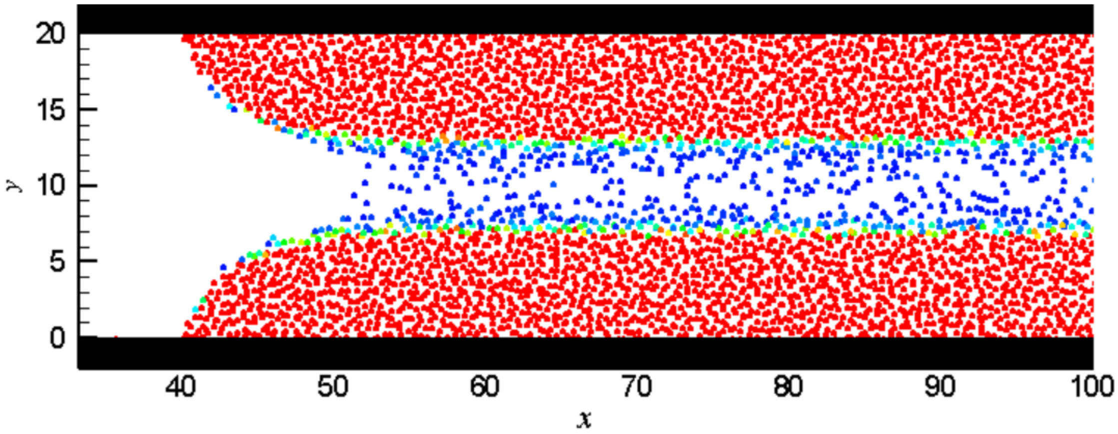
(c)

(d)



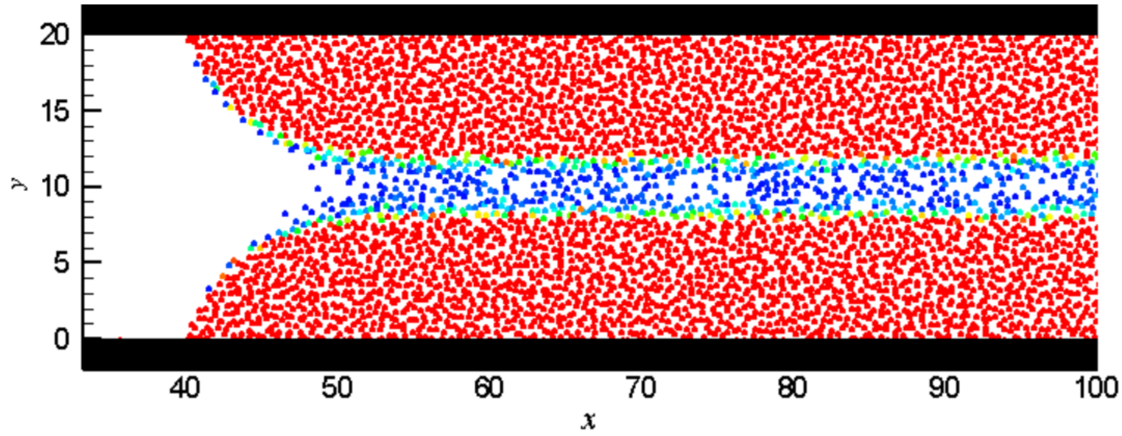
20
15
y
10
5
0
40
50
60
70
80
90
100
x

(e)



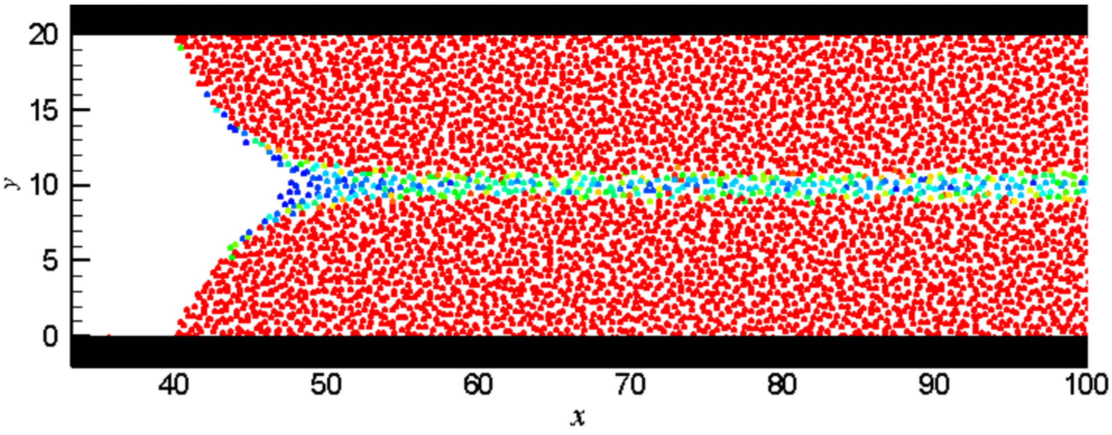
20
15
y
10
5
0
40
50
60
70
80
90
100
x

(f)

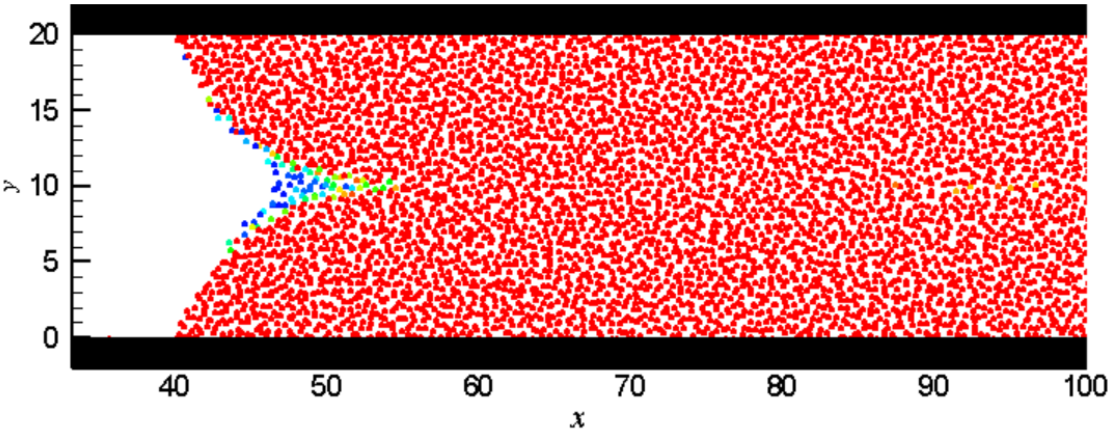


20
15
y
10
5
0
40
50
60
70
80
90
100
x

(g)



(h)

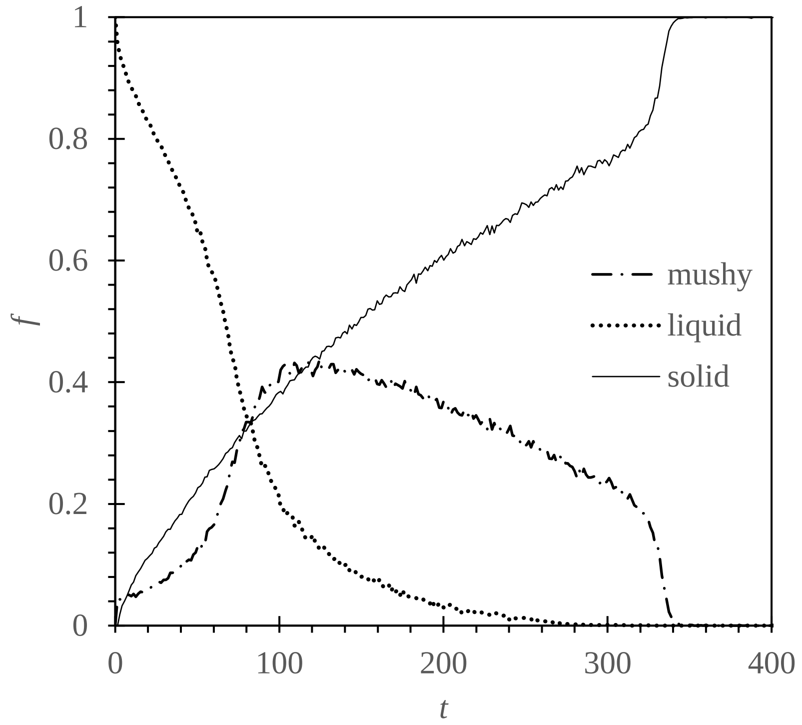


(i)

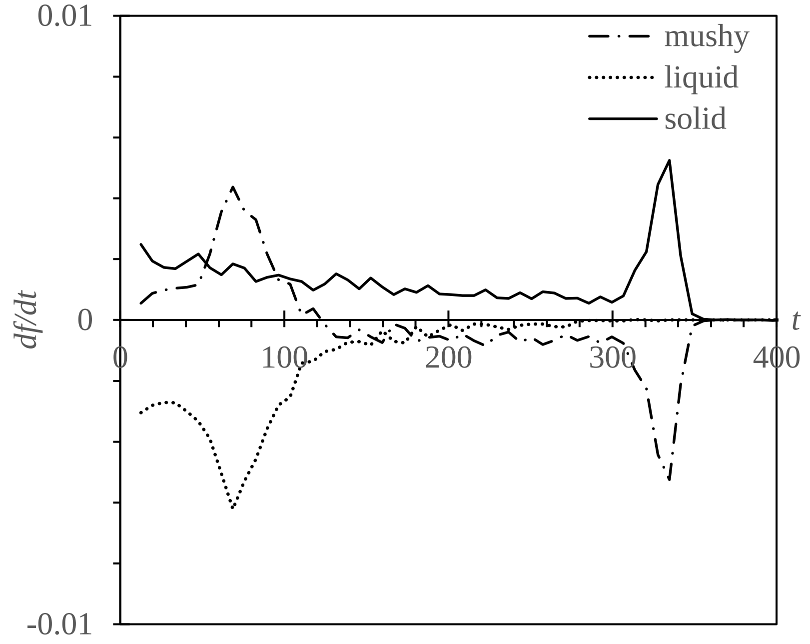
Figure 27: Solidification process in the flow channel for the intense cooling case: T_{w1} = 0.85. Only solid (red) and mushy particles are displayed. (a) t = 1.4; (b) t = 7.0; (c) t = 21; (d) t = 29.4; (e) t = 35; (f) t = 40.6; (g) t = 50.4; (h) t = 61.6; (i) t = 67.2.

On the other hand, when the wall is subjected to mild cooling (T_{w1} = 0.90), f_M is increasing soon after the cooling starts (Figure 28(a)), owing to the fact that the number of liquid particles that turn into the transition state is more than that of the mushy particles that turn into the solid state. In fact, at t<100, the growth rate of f_M is almost on par with the decay rate of f_L as seen in Figure 28(b), signifying that the formation rate of solid layers near the walls are rather mild initially. The number of mushy particles peaks at t~100. Beyond t=100, the decay rate of f_L slows down gradually until all liquid particles are no longer visible at t~300. Meanwhile, f_M is descending as well at a consistent rate within 200<t<300 as shown in Figure 28(b). At t~300 (f_L = 0), still, almost a quarter of the region considered consist of mushy particles (f_M ~ 25%). This observation is in contrast with the intense cooling case, where the mushy particles occupy only 10% of the region as f_L = 0. At t>300, since liquid particles are no longer available, f_M undergoes a sharp decrease as more mushy particles are now turning into solids. Full solid region is formed at t~350.

Johansson et al. (2016) argued that the channel closing time is longer if T_{w1} approaches the free-stream temperature caused by the surface roughness between the solid-liquid interface. The increase of surface roughness, according to them, is due to the freezing/unfreezing behavior of fluid particles. In fact, we have noticed similar behavior in the current computation. Figure 29 shows the time evolutions of f_L and f_M for different T_{w1}. As seen, fluctuations in f_M are more discernible as T_{w1} increases, signifying that liquid particles may undergo freezing/unfreezing if milder cooling (high T_{w1}) is performed. This process will prolong the channel closing time (t_c), as shown in Figure 30. In general, t_c increases almost linearly with respect to wall temperature when more intense cooling is performed (low T_{w1}). However, as T_{w1} increases, t_c increases with the wall temperature in a non-linear fashion. This observation is agreeable with that reported by Johansson et al. (2016).

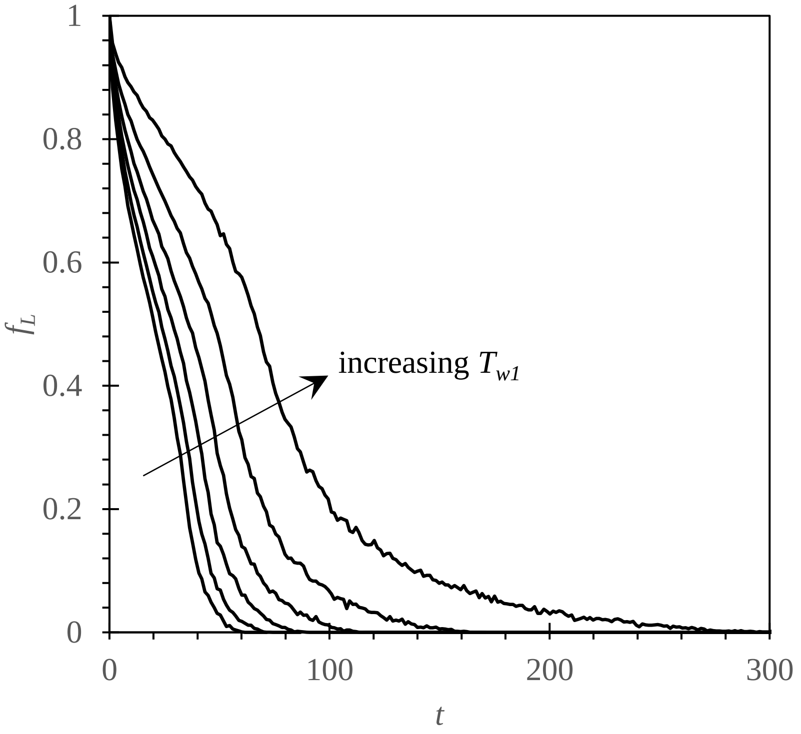


(a)

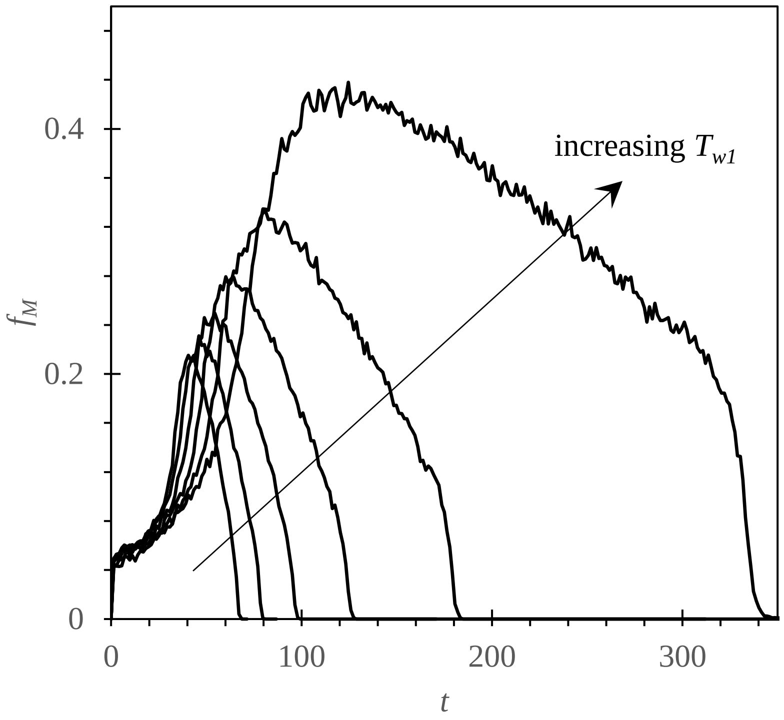


(b)

Figure 28: Time histories of (a) phase factions and (b) rates of change of phase fractions within the region: $70 < x < 100$. $T_{w1} = 0.90$.



(a)



(b)

Figure 29: Time evolutions of (a) liquid fraction and (b) mushy fraction for T_{w1} = 0.85, 0.86, 0.87, 0.88, 0.89 and 0.90 within the region: $70 < x < 100$.

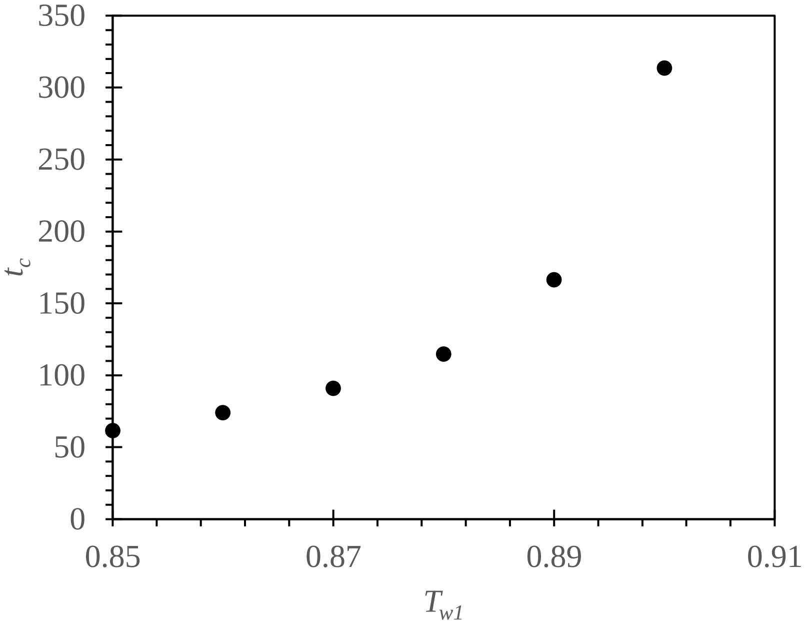


Figure 30: Channel closing times for various wall temperatures T_{w1}.

Conclusions

In the current work, a more systematic method has been proposed in order to predict the temperature-dependent function of the exponent s appeared in the weight function of the dissipative force term. We have found that the selection of cut-off radius (r_c) is highly dependent on the temperature range considered. The thermophysical properties of water including Schmidt and Prandtl numbers have been well reproduced within the temperature range considered in the current work: $253.15\ \text{K} < T^R < 373.15\text{K}$.

A variety of flow problems involving solidification have been studied. In order to properly simulate the speed of the solid-liquid interface, mushy particles have been used to model the eDPD particles within the transition zone. The effective thermal diffusivity of the mushy particles is then 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 solid varies considerably from that of liquid.

Also, we have attempted a new 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 generate the mirror particles (of the ghost particles in the wall layer) inside the flow domain once, and the temperatures of these mirror particles are interpolated from the neighbouring 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 validated the eDPD method in 2D solidification problem which has been overlooked so far.

Finally, the freezing problem in a flow channel has been computed and it has been found that the closing time of the channel increases linearly when intense cooling is performed (T_{w1}<0.87). Beyond this critical temperature (milder cooling), the closing time increases with the wall temperature in a non-linear fashion. We have observed that the fluctuations of mushy fraction f_M are more discernible as mild cooling is performed, signifying that liquid particles indeed undergo frequent freezing/unfreezing during the solidification process. This would prolong the channel closing time undoubtedly.

References

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

References

[1.] Abu-Nada, E. (2015), “Dissipative particle dynamics simulation of natural convection using variable thermal properties”, International Communications in Heat and Mass Transfer, Vol. 69, pp. 84-93.

[2.] Alexiades, V. and Solomon, A.D. (1993), “Mathematical modeling of melting and freezing processes”, Hemisphere Publishing Corporation, 323 pages.

[3.] Backer, J.A., Lowe, C.P., Hoefsloot, H.C. and Iedema, P.D. (2005), “Poiseuille flow to measure the viscosity of particle model fluids”, The Journal of Chemical Physics, Vol. 122, 154503.

[4.] Chaudhri, A. and Lukes, J.R. (2009), “Multicomponent energy conserving dissipative particle dynamics: A general framework for mesoscopic heat transfer applications”, Journal of Heat Transfer, Vol. 131, No. 3, 033108.

[5.] Dhanabalan, S.S. (2005), “On the boundary conditions for dissipative particle dynamics’, Master Thesis, National University of Singapore, 115 pages.

[6.] Duong-Hong, D., Phan-Thien, N., Yeo, K.S. and Ausias, G. (2010), “Dissipative particle dynamics simulations for fibre suspensions in Newtonian and viscoelastic fluids”, Computer Methods in Applied Mechanics and Engineering, Vol. 199, pp. 1593-1602.

[7.] Espanol, P. (1997), “Dissipative particle dynamics with energy conservation”, Europhysics Letter, Vol. 30, No. 4, pp. 191-196.

[8.] Fan, X., Phan-Thien, N., Ng, T.Y., Wu, X. and Xu, D. (2003), “Microchannel flow of a macromolecular suspension”, Physics of Fluids, Vol. 15, No. 1, pp. 11-21.

[9.] Fan, X., Phan-Thien, N., Chen, S., Wu, X. and Ng, T.Y. (2006), “Simulating flow of DNA suspension using dissipative particle dynamics”, Physics of Fluids, Vol. 18, 063102.

[10.] Fedosov, D.A., Sengupta, A. and Gompper, G. (2015), “Effect of fluid-colloid interactions on the mobility of a thermophoretic microswmmer in non-ideal fluids”, Soft Matter, Vol. 11, pp. 6703-6715.

[11.] Ge, S., Gu, Y. and Chen, M. (2015), “A molecular dynamics simulation on the convective heat transfer in nanochannels”, Molecular Physics, Vol. 113, pp. 703-710.

[12.] Gillen, K.T., Douglass, D.C. and Hoch, M.J.R. (1972), “Self-diffusion in liquid water to -31^{o}C”, The Journal of Chemical Physics, Vol. 57, No. 12, pp. 5117-5119.

[13.] Groot, R.D. and Warren, P.B. (1997), “Dissipative Particle Dynamics: Bridging the gap between atomistic and mesoscopic simulation”, Journal of Chemical Physics, Vol. 107, No. 11, pp. 4423-4435.

[14.] Guo, J., Li, X., Liu, Y. and Liang, H. (2011), “Flow-induced translocation of polymers through a fluidic channel: A dissipative particle dynamics simulation study”, The Journal of Chemical Physics, Vol. 134, 134906.

[15.] Hallett, J. (1963), “The temperature dependence of the viscosity of supercooled water”, Proceedings of the Physical Society, Vol. 82, No. 6, pp. 1046-1050.

[16.] Hinz, D.F., Panchenko, A., Kim, T.Y. and Fried, E. (2015), “Particle-based simulations of self-motile suspensions”, Computer Physics Communications, Vol. 196, pp. 45-57.

[17.] Holten, V., Bertrand, C.E., Anisimov, M.A. and Sengers, J.V. (2012), “Thermodynamics of supercooled water”, Journal of Chemical Physics, Vol. 136, 094507.

[18.] Hoogerbrugge, P.J. and Koelman, J.M.V.A. (1992), “Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics”, Europhysics Letters, Vol. 19, No. 3, pp. 155-150.

[19.] Johansson, E.O., Yamada, T., Sunden, B. and Yuan, J. (2016), “Modeling mesoscopic solidification using dissipative particle dynamics”, International Journal of Thermal Sciences, Vol. 101, pp. 207-216.

[20.] Kumar, A. (2009), “Computational modeling of microfluidic processes using dissipative particle dynamics”, Ph.D. Thesis, University of Rhode Island, USA, 138 pages.

[21.] Li, Z. and Drazer, G. (2008), “Hydrodynamics interactions in dissipative particle dynamics”, Physics of Fluids, Vol. 20, 103601.

[22.] Li, Z., Tang, Y.H., Lei, H., Caswell, B. and Karniadakis, G.E. (2014), “Energy-conserving dissipative particle dynamics with temperature-dependent properties”, Journal of Computational Physics, Vol. 265, pp. 113-127.

[23.] Liu, M.B., Liu, G.R., Zhou, L.W. and Chang, J.Z. (2014), “Dissipative particle dynamics (DPD): An overview and recent developments”, Archives of Computational Methods in Engineering, Vol. 22, Issue 4, pp. 529-556.

[24.] Lunardini, V.J. (1988), “Heat conduction with freezing of thawing”, Monograph 88-1, U.S. Army Cold Regions Research and Engineering Laboratory, 344 pages.

[25.] Mahmudah, R.S., Kumabe, M., Suzuki, T., Guo, L. and Morita, K. (2011), “Particle-based simulations of molten metal flows with solidification”, Memoirs of the Faculty of Engineering, Kyushu University, Vol. 71, No. 1, pp. 17-29.

[26.] Mai-Duy, N., Pan, D., Phan-Thien, N. and Khoo, B.C. (2013), “Dissipative particle dynamics modeling of low Reynolds number incompressible flows”, Journal of Rheology, Vol. 57, pp. 585-604.

[27.] Mills, Z.G., Mao, W. and Alexeev, A. (2013), “Mesoscale modeling: solving complex flows in biology and biotechnology”, Trends in Biotechnology, Vol. 31, No. 7, pp. 426-434.

[28.] Moeendarbary, E., Lam, K.Y. and Ng, T.Y. (2008), “A new bounce-normal boundary in DPD calculations for the reduction of density fluctuations”, Proceedings of MNHT2008, Micro/Nanoscale Heat Transfer International Conference, January 6-9, 2008, Tainan, Taiwan, Paper MNHT2008-52069, pp. 1-10.

[29.] Moeendarbary, E., Ng, T.Y. and Zangeneh, M. (2009), “Dissipative particle dynamics: introduction, methodology, and complex fluid applications-A review”, Vol. 1, No. 4, pp. 737-763.

[30.] Ng, K.C., T.W.H., Sheu and Hwang, Y.H. (2016), “Unstructured Moving Particle Pressure Mesh (UMPPM) method for incompressible isothermal and non-isothermal flow computation”, Computer Methods in Applied Mechanics and Engineering, Vol. 305, pp. 703-738.

[31.] Qiao, R. and He, P. (2007), “Simulation of heat conduction in nanocomposite using energy-conserving dissipative particle dynamics”, Molecular Simulation, Vol. 33, No. 8, pp. 677-683.

[32.] Rathjen, K.A. and Jiji, L.M. (1971), “Heat conduction with melting or freezing in a corner”, Journal of Heat Transfer, Vol. 93, Issue 1, pp. 101-109.

[33.] Ripoll, M., Mussawisade, K., Winkler, R.G. and Gompper, G. (2005), “Dynamic regimes of fluids simulated by multiparticle-collision dynamics”, Physical Review E, Vol. 72, 016701.

[34.] Steiner, T., Cupelli, C. and Zengerle, R. (2009), “Simulation of advanced microfluidic systems with dissipative particle dynamics”, Microfluid Nanofluid, Vol. 7, pp. 307-323.

[35.] Symeonidis, V., Karniadakis, G.E. and Caswell, B. (2006), “Schmidt number effects in dissipative particle dynamics simulation of polymers”, Journal of Chemical Physics, Vol. 125, 184902.

[36.] Tang, Y.H., Li, Z., Li, X., Deng, M. and Karniadakis, G.E. (2016), “Non-equilibrium dynamics of vesicles and micelles by self-assembly of block copolymers with double thermoresponsivity”, Macromolecules, Vol. 49, No. 7, pp. 2895-2903.

[37.] Willemsen, S.M., Hoefsloot, H.C.J., Visser, D.C., Hamersma, P.J. and Iedema, P.D. (2000), “Modelling phase change with dissipative particle dynamics using a consistent boundary condition”, Journal of Computational Physics, Vol. 162, pp. 385-394.

[38.] Willemsen, S.M., Hoefsloot, H.C.J. and Iedema, P.D. (2002), “Mesoscopic simulation of polymers in fluid dynamics problems”, Journal of Statistical Physics, Vol. 107, Nos 1/2, pp. 53-65.

[39.] Yamada, T., Johansson E.O., Sunden, B. and Yuan, J. (2016), “Dissipative particle dynamics simulations of water droplet flows in a submicron parallel-plate channel for different temperature and surface-wetting conditions”, Numerical Heat Transfer, Part A, Vol. 70, Issue 6, pp. 595-612.

[40.] Zhang, Y.X., Yi, H.L. and Tan, H.P. (2016), “A dissipative particle dynamics algorithm for fluid-solid conjugate heat transfer”, International Journal of Heat and Mass Transfer, Vol. 103, pp. 555-563.