# A dissipative particle dynamics method for arbitrarily complex geometries

Zhen Li [a,*], Xin Bian [b], Yu-Hang Tang [a], George Em Karniadakis [a,*]

[a] *Division of Applied Mathematics, Brown University, Providence, RI, 02912, USA*
[b] *Department of Mechanical Engineering, Technical University of Munich, Boltzmannstr. 15, 85748 Garching, Germany*

## ARTICLE INFO

## ABSTRACT

Dissipative particle dynamics (DPD) is an effective Lagrangian method for modeling complex fluids in the mesoscale regime but so far it has been limited to relatively simple geometries. Here, we formulate a local detection method for DPD involving arbitrarily shaped geometric three-dimensional domains. By introducing an indicator variable of boundary volume fraction (BVF) for each fluid particle, the boundary of arbitrary-shape objects is detected on-the-fly for the moving fluid particles using only the local particle configuration. Therefore, this approach eliminates the need of an analytical description of the boundary and geometry of objects in DPD simulations and makes it possible to load the geometry of a system directly from experimental images or computer-aided designs/drawings. More specifically, the BVF of a fluid particle is defined by the weighted summation over its neighboring particles within a cutoff distance. Wall penetration is inferred from the value of the BVF and prevented by a predictor–corrector algorithm. The no-slip boundary condition is achieved by employing effective dissipative coefficients for liquid–solid interactions. Quantitative evaluations of the new method are performed for the plane Poiseuille flow, the plane Couette flow and the Wannier flow in a cylindrical domain and compared with their corresponding analytical solutions and (high-order) spectral element solution of the Navier–Stokes equations. We verify that the proposed method yields correct no-slip boundary conditions for velocity and generates negligible fluctuations of density and temperature in the vicinity of the wall surface. Moreover, we construct a very complex 3D geometry – the "Brown Pacman" microfluidic device – to explicitly demonstrate how to construct a DPD system with complex geometry directly from loading a graphical image. Subsequently, we simulate the flow of a surfactant solution through this complex microfluidic device using the new method. Its effectiveness is demonstrated by examining the rich dynamics of surfactant micelles, which are flowing around multiple small cylinders and stenotic regions in the microfluidic device without wall penetration. In addition to stationary arbitrary-shape objects, the new method is particularly useful for problems involving moving and deformable boundaries, because it only uses local information of neighboring particles and satisfies the desired boundary conditions on-the-fly.

---

* Corresponding authors.
  *E-mail addresses:* zhen_li@brown.edu (Z. Li), george_karniadakis@brown.edu (G.E. Karniadakis).

## 1. Introduction

Despite of the sustained fast growth of computing power during the past few decades, it is still computationally prohibitive or impractical to model long time scales and large spatial scales in many applications of soft matter and biological systems with the brute-force atomistic simulations [1,2]. If only the mesoscopic properties and collective behavior are of practical interest, it may not be necessary to explicitly take into account all the details of materials at the atomic/molecular level [3]. To this end, a coarse-graining approach eliminates fast degrees of freedom and drastically simplifies the dynamics on atomistic scales, while providing a cost-effective simulation path to capturing the correct properties of complex fluids at larger spatial and temporal scales beyond the capacity of conventional atomistic simulations [4]. In recent years, with increasing attention on the research of soft matter and biophysics [5], coarse-grained (CG) modeling has become a rapidly expanding methodology especially in the simulations of polymers [6–9], colloidal suspensions [10–12], interfaces of multiphase fluids [13–15], cell dynamics [16–18], blood rheology [19–21] and biological materials [22–25].

Initially proposed by Hoogerbrugge and Koelman [26], dissipative particle dynamics (DPD) is one of the currently most popular CG methods [27–29] for performing mesoscopic simulations of complex fluids. The DPD particles are defined as coarse-grained entities [30,31], which represent clusters of molecules rather than atoms/molecules directly. In contrast to molecular dynamics (MD) method, DPD allows much larger particle size and also time steps because of the soft particle interactions. As a particle-based mesoscopic method, DPD considers $N$ particles, whose state variables of momentum and position are governed by the Newton's equations of motion [32]. For a typical DPD particle $i$, its time evolution follows $\dot{\mathbf{r}}_i = \mathbf{v}_i$ and $\dot{\mathbf{p}}_i = \mathbf{F}_i = \sum_{i \neq j}(\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R)$ where $\mathbf{r}_i$, $\mathbf{v}_i$, $\mathbf{p}_i$ and $\mathbf{F}_i$ denote position, velocity, momentum and force vectors, respectively. The summation for computing the total force $\mathbf{F}_i$ is carried out over all other particles within a cutoff radius $r_c$ beyond which the forces are considered negligible. The pairwise force $\mathbf{F}_{ij}$ comprises conservative ($\mathbf{F}_{ij}^C$), dissipative ($\mathbf{F}_{ij}^D$) and random ($\mathbf{F}_{ij}^R$) forces, which are expressed as [32]

$$
\begin{aligned}
\mathbf{F}_{ij}^C &= a_{ij}\omega_C(r_{ij})\mathbf{e}_{ij}, \\
\mathbf{F}_{ij}^D &= -\gamma_{ij}\omega_D(r_{ij})(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij}, \\
\mathbf{F}_{ij}^R &= \sigma_{ij}\omega_R(r_{ij})\mathrm{d}\tilde{W}_{ij}\mathbf{e}_{ij},
\end{aligned}
\tag{1}
$$

where $r_{ij} = |\mathbf{r}_{ij}| = |\mathbf{r}_i - \mathbf{r}_j|$ represents the distance between two particles $i$ and $j$, $\mathbf{e}_{ij} = \mathbf{r}_{ij}/r_{ij}$ is the unit vector from particles $j$ to $i$, and $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ is the velocity difference; $\mathrm{d}\tilde{W}_{ij}$ is an independent increment of the Wiener process [33]. Also, $\gamma_{ij}$ is the dissipative coefficient and $\sigma_{ij}$ sets the strength of random force. The dissipative force and random force together act as a thermostat when the dissipative coefficient $\gamma$ and the amplitudes of white noise $\sigma$ satisfy the fluctuation–dissipation theorem (FDT) [33,34] requiring $\sigma^2 = 2\gamma k_B T$ and $\omega_D(r) = \omega_R^2(r)$. All these forces in Eq. (1) have the same finite interaction range $r_c$ and their amplitudes decay according to corresponding weight functions. A common choice of the weight functions [32] is $\omega_C(r) = 1 - r/r_c$ and $\omega_D(r) = \omega_R^2(r) = (1 - r/r_c)^2$ for $r \leq r_c$ and zero for $r > r_c$.

All the three forces between DPD particles are soft and short-range interactions, which allow large time steps for the time integration of the particle-based system. The soft interactions between DPD particles, unlike the hard potentials in atomistic simulations, cannot prevent fluid particles from penetrating wall boundaries [35]. It is also unlike the top–down smoothed particle hydrodynamics (SPH) [36] or smoothed DPD (SDPD) [10] approach, where the equation of state can be tuned so that the pressure is arbitrarily strong to prevent particle penetration. As a result, for wall-bounded flow systems, DPD simulations require extra formulations [37–39] to prevent the penetration of the liquid particles into solid boundaries. Specular, Maxwellian, and bounce-back reflections [40] are common techniques used to reflect particles back into the fluid after they cross the wall surface. Therefore, for wall-bounded flows one has to mathematically predefine the position of solid wall to judge the penetration of fluid particles before a DPD simulation can be performed, which is difficult to extend for arbitrarily shaped boundaries and limits the applicability of DPD.

In the present paper, we develop a boundary method for imposing correctly the no-slip boundary condition on the solid walls with arbitrary shapes. Instead of predefining the position of the wall boundary, we make the fluid particles autonomous to detect the wall surface and to infer the wall penetration by themselves based on the local information of their neighboring particles. Hence, the geometry of solid boundary can be computed on-the-fly using local particle configurations. Therefore, it is no longer necessary to predefine the boundary geometry for DPD simulations, which makes it possible to construct DPD systems with arbitrary-shape domains directly from loading experimental images or computer-aided designs/drawings. Furthermore, since this boundary method uses local information of neighboring particles and satisfies no-slip/partial-slip boundary conditions on-the-fly, it is not only valuable for stationary arbitrary-shape boundaries but also for moving boundaries and deformable boundaries.

The remainder of this paper is organized as follows: Section 2 introduces the details of the boundary method, and also how to compute the effective dissipative coefficient for liquid–solid interactions. In Section 3, we validate the proposed boundary method by performing the Poiseuille flow, the Couette flow and the Wannier flow with comparison to analytical solutions. Moreover, an error analysis of this boundary method related to the curvature of arbitrary-shaped boundaries is provided in Appendix A. We also include a demonstration of micelles flowing through a very complex microfluidic device. Finally, we end with a brief summary and discussion in Section 4.
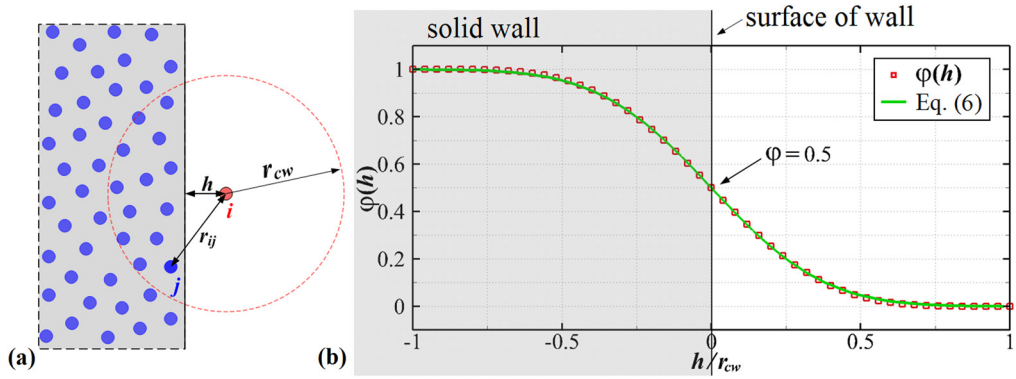
**Fig. 1.** (a) Sketch for a fluid particle $i$ in the vicinity of a wall represented by many solid particles and the integration domain for accumulation of $\phi$. (b) Dependence of $\phi(h)$ on $h/r_{cw}$ calculated by the integration in Eq. (4).

## 2. Wall boundary method

### 2.1. Definition of the boundary volume fraction

Consider a fluid particle $i$ in the vicinity of a solid wall represented by discrete DPD particles and we assign to it an extra variable $\phi_i = \varphi(\mathbf{r}_i)$ in addition to other quantities such as position and momentum. We define $\phi_i$ as the boundary volume fraction (BVF) depending on the coordinates of particle $i$. More specifically, the value of $\phi_i$ is computed using a weighted summation over neighboring solid particles $j$ given by

$$\phi_i = \varphi(\mathbf{r}_i) = \frac{1}{\rho_w} \sum_{\substack{j \in \mathbf{S} \\ r_{ij} < r_{cw}}} W(r_{ij}, r_{cw}), \tag{2}$$

where $W(r, r_{cw})$ is a weighting function, and $\rho_w$ is the bulk number density of solid particles. The weighting function $W(r, r_{cw})$ can be any smoothing kernel, such as the ones used widely in smoothed particle hydrodynamics [41,42]. As a demonstration, we choose the three-dimensional Lucy kernel [43]

$$W(r, r_{cw}) = \frac{105}{16\pi r_{cw}^3} \left(1 + \frac{3r}{r_{cw}}\right) \left(1 - \frac{r}{r_{cw}}\right)^3, \tag{3}$$

where $r$ is the norm of $\mathbf{r}$, and $r_{cw}$ is the cutoff radius beyond which $W(r, r_{cw})$ is considered zero. Larger $r_{cw}$ increases the computational cost but yields smoother $\varphi(\mathbf{r})$, as we will discuss in section 3. Unless otherwise specified, in testing cases we simply set $r_{cw}$ equal to $r_c$.

Consider a planar wall surface or a wall surface with a radius of curvature far greater than the cutoff radius $r_{cw}$, as shown in Fig. 1(a); we estimate the value of $\phi_i$ using the continuum approximation

$$\phi_i = \varphi(h) = \int\limits_{z=h}^{r_{cw}} \int\limits_{x=0}^{\sqrt{r_{cw}^2 - z^2}} 2\pi x W(r, r_{cw}) \cdot dx \cdot dz \tag{4}$$

where $r = \sqrt{x^2 + z^2}$ and $h$ is the distance between the particle $i$ and the wall boundary. By inserting the Lucy kernel given by Eq. (3) into Eq. (4), we have

$$\varphi(h) = \begin{cases} \frac{1}{16}\left(1 - \frac{h}{r_{cw}}\right)^5 \left(15\left(\frac{h}{r_{cw}}\right)^2 + 19\frac{h}{r_{cw}} + 8\right) & 0 \leqslant h \leqslant r_{cw}, \\ 1 - \varphi(-h) & -r_{cw} \leqslant h < 0 \end{cases} \tag{5}$$

in which $\varphi(h = 0) = 0.5$. It is worth noting that $h = 0$ represents that the particle lies right on the wall surface, and a negative $h$ means that the particle is underneath the wall surface while a positive $h$ is for the particle outside the wall boundary. The number density $\rho_w$ disappears in Eq. (4) because we scaled $\varphi(\mathbf{r})$ by $\rho_w$ in Eq. (2). Thus, the value of $\phi_i$ only depends on $h$ as plotted in Fig. 1(b), which shows clearly that $\varphi(h)$ decreases from 1.0 to 0 as $h$ changes from $-r_{cw}$ to $r_{cw}$.

Given a value of $\phi_i$, the distance of a particle $i$ away from the wall surface can be computed by the inverse function of Eq. (5). For simplifying the numerical implementation in practical simulations, we employ an approximation of $\varphi^{-1}(\phi_i)$ in the form of

$$h_i/r_{cw} = \varphi^{-1}(\phi_i) \approx \begin{cases} 1 - \left(2.088\phi_i^3 + 1.478\phi_i\right)^{1/4} & 0 \leqslant \phi_i \leqslant 0.5 , \\ -\varphi^{-1}(1 - \phi_i) & 0.5 < \phi_i \leqslant 1. \end{cases} \tag{6}$$

The value of BVF on each fluid particle $\phi_i$ can be evaluated every time step in the same loop of pairwise force computation, so the extra computational cost for applying this boundary method is marginal. Given a value of $\phi_i$, the distance of the particle $i$ from the wall surface is given by $h = \varphi^{-1}(\phi_i) \cdot r_{cw}$. Whenever $h_i < 0$ (or $\phi_i > 0.5$), the wall penetration is observed.

For a fluid DPD particle $i$ with state variables $\{\mathbf{r}_i, \mathbf{v}_i\}$, we employ a predictor–corrector algorithm to prevent the wall penetration. In particular, we perform an imaginary-integration of its position for a time step $\Delta t$, i.e., $\mathbf{r}'_i = \mathbf{r}_i + \mathbf{v}_i \Delta t$. If the value $\varphi(\mathbf{r}'_i) > 0.5$, the particle $i$ at the predicted position $\mathbf{r}'_i$ would penetrate into the wall. To avoid this wall penetration, we correct the velocity of the particles whose $\varphi(\mathbf{r}'_i)$ greater than 0.5 by reassigning a new value

$$\mathbf{v}_i^{\text{new}} = 2\mathbf{U} + \mathbf{a}\Delta t - \mathbf{v}_i + 2 \max\{0, \mathbf{v}_i \cdot \mathbf{e}_n\} \cdot \mathbf{e}_n, \tag{7}$$

where $\mathbf{e}_n$ is the unit normal vector of the wall boundary, $\mathbf{U}$ and $\mathbf{a}$ are the local velocity and local acceleration of the boundary, respectively. It is obvious that all stationary wall boundaries have $\mathbf{U} = \mathbf{a} = 0$. For moving boundaries, the value of $\mathbf{U}$ and $\mathbf{a}$ can take the values of velocity and acceleration of the nearest wall particle in practical DPD simulations. Let $\mathbf{n}_w$ be the gradient of $\varphi(\mathbf{r}_i)$ at the location of particle $i$, which is computed by

$$\mathbf{n}_w = \nabla\varphi(\mathbf{r}_i) = \frac{1}{\rho_w} \sum_{\substack{r_{ij} < r_{cw}}}^{j \in \mathbf{S}} \frac{\mathbf{r}_{ij}}{r_{ij}} \frac{dW(r_{ij}, r_{cw})}{dr_{ij}}. \tag{8}$$

Then, the unit normal vector of the wall boundary $\mathbf{e}_n = \mathbf{n}_w/n_w$ where $n_w$ is the modulus of $\mathbf{n}_w$.

### 2.2. Control of the surface roughness

Consider a flat wall represented by solid particles in uniform lattices, the iso-surface of $\phi = 0.5$ is smooth and flat, which can accurately represent the surface of the flat wall. However, the structure of solid particles associated with these lattices will induce unwanted fluctuations [37] of fluid density and temperature in the vicinity of wall boundary. There are different methods to eliminate the wall-induced fluctuations in the bulk region, such as an adaptive model for wall-particle interactions [44], an amorphous wall model [45] and a dynamic virtual particle allocation method [46]. In the present paper, we employ randomly distributed particles in the wall domain to represent the wall boundary, which is easier and more general for construction of fluid system with complex geometry and arbitrarily shaped boundaries.

Randomly distributed particles do not possess a lattice structure, and hence effectively eliminate the fluctuations of averaged profiles of density and temperature in the vicinity of the wall surface. Theoretically, to the limit of the continuum approximation, a planar wall surface can be accurately represented by the isosurface of $\phi = 0.5$ when the solid particles are dense enough. However, in practical implementations, the number density $\rho_w$ in DPD simulations is finite and usually smaller than 10.0. As a result, the roughness on wall surface is generated by the random distribution of discrete particles with finite number density. Figs. 2 shows the wall boundary represented by the isosurface of $\phi = 0.5$ for the number densities of wall particles being $\rho_w = 4.0$ and 8.0. The solid wall is made up of randomly distributed particles, where the average distance between these particles is $\delta = \rho_w^{-1/3} = 0.63$ for $\rho_w = 4.0$ and $\delta = 0.5$ for $\rho_w = 8.0$.

The surface roughness shown in Fig. 2 comes from the estimation error of $\phi$ based on the discrete particles. Unlike SPH or other mesh-free methods that try to eliminate this error, we take advantage of this kind of error for generating controllable roughness on wall surface in the DPD systems. As a matter of fact, any natural solid wall contains more or less chemical/physical heterogeneities on the surface especially at the mesoscopic scale. Such heterogeneity can be modeled qualitatively by the roughness on walls in the DPD systems, as shown in Fig. 2. More importantly, the proposed boundary method provides a convenient way to generate various sizes of the roughness for representation of different degrees of the heterogeneity.

According to the unbiased estimation of standard deviation [47], the magnitude of roughness associated with the randomness of particle distribution will monotonically decrease as the number of neighboring particles $N_w = 2\pi r_{cw}^3/3 \cdot \rho_w$ increases, implying that the roughness of the wall is controllable by tuning the cutoff radius $r_{cw}$ and number density $\rho_w$ of DPD systems. To quantify the wall surface texture, we define the root mean squared height $R_q$ given by

$$R_q = \left(\frac{1}{A} \iint \Delta^2(x, y) dx dy\right)^{1/2} \tag{9}$$

where $A$ is the area of a flat wall, and $\Delta(x, y)$ represents the vertical deviations of a real surface of $\phi = 0.5$ from its ideal surface defined by the isosurface of $\phi = 0.5$ when $\rho_w \to \infty$. Fig. 3 shows that the root mean squared height $R_q$ decreases as the cutoff radius $r_{cw}$ and the number density $\rho_w$ increase. Since $r_{cw}$ is only involved in computation of $\phi$, the variation of $r_{cw}$ does not affect the particle interactions and fluid properties. In practical DPD simulations, changing the value of $N_w^{-1/2}$ allows linear tunability of the wall surface roughness, as shown in Fig. 3(b).
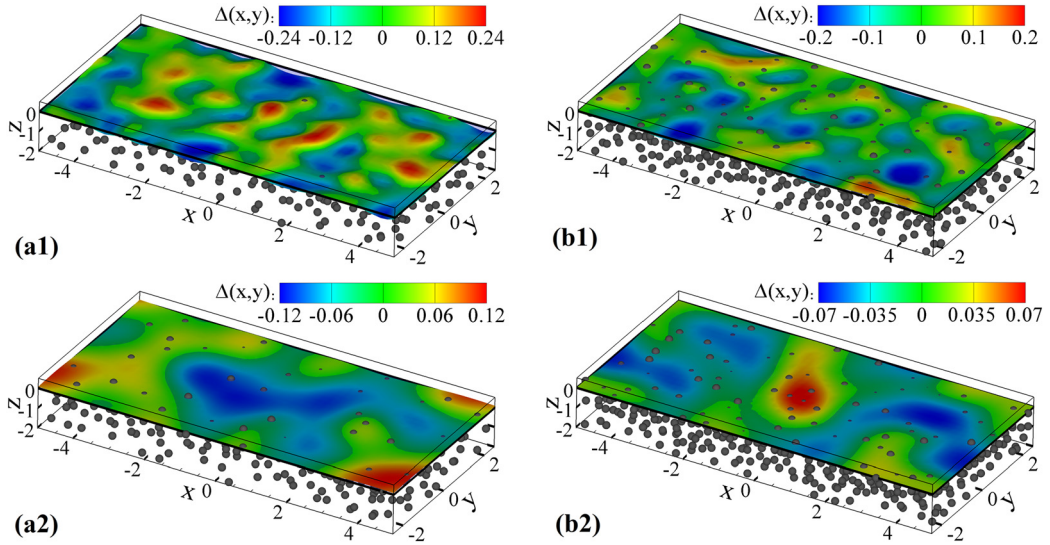
**Fig. 2.** Wall boundary represented by the isosurface of $\phi = 0.5$ computed by Eq. (2) for a number density of wall particles $\rho_w = 4.0$ with the cutoff radius of (a1) $r_{cw} = 1.0$ and (a2) $r_{cw} = 2.0$, and for a number density $\rho_w = 8.0$ with the cutoff radius of (b1) $r_{cw} = 1.0$ and (b2) $r_{cw} = 2.0$. The value of $\Delta(x, y)$ represents the vertical deviations of an isosurface of $\phi = 0.5$ from its ideal surface $z = 0$. Spherules represent the randomly distributed solid particles.
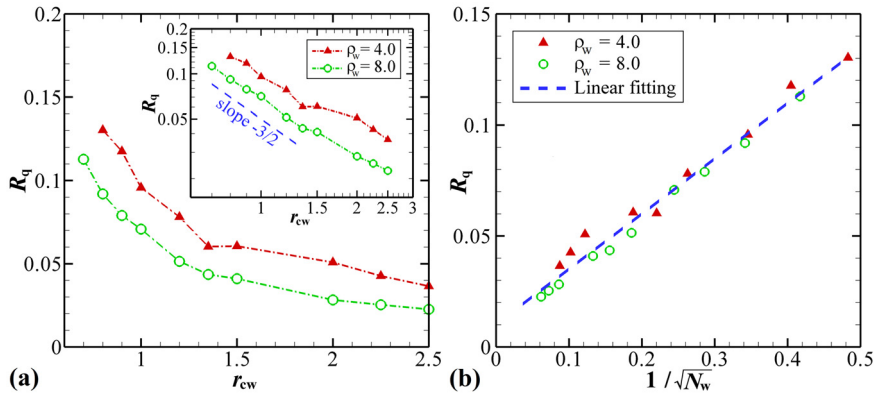


**Fig. 3.** (a) Dependence of the root mean squared height $R_q$ on the cutoff radius $r_{cw}$, where the inset shows the scaling law of $R_q \sim r_{cw}^{-3/2}$. (b) The scaling law of $R_q \propto 1/\sqrt{N_w}$ indicating the linear tunability of the wall surface roughness, where $N_w$ is the number of neighboring solid particles on wall surface, i.e., $N_w = 2\pi r_{cw}^3 / 3 \cdot \rho_w$.

Usually, a randomized configuration of DPD particles can be generated simply by a random number generator, which may result in overlapping of particles or large vacancies in wall boundaries. To avoid the overlaps or vacancies, a more uniform particle distribution is needed, which can be achieved by a process of geometry optimization or a short run of particle-based simulation. In the present paper, we carry out a short DPD simulation with a relatively large conservative force coefficient to get the initial particle positions. Then, the particles in the wall domain are frozen as solid particles, while others in the fluid domain are taken as the fluid particles. For instance, the wall boundary shown in Fig. 2 is obtained by running 1000 time steps of DPD simulation from totally randomized particles.

## 2.3. Effective dissipative interaction

Applying the bounce-back reflection without any correction on dissipative interactions will violate the no-slip condition on wall surface in DPD simulations. This is a widely discussed issue for DPD and it is the motivation of many previous DPD works [37,48–50] for imposing correct no-slip boundary conditions at wall surface. In this section, we will introduce a method to implement an effective dissipative interaction, so that the local detection method is able to impose correct no-slip boundary conditions at wall boundaries.

The dissipative force between two DPD particles is computed by $\mathbf{F}_{ij}^D = -\gamma \cdot \omega_D(r_{ij})(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij}$, where $r_{ij}$ is the distance between particles $i$ and $j$, $\mathbf{e}_{ij}$ represents the unit vector from particle $j$ to $i$, and $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ is their velocity difference. The weighting function is given by $\omega_D(r_{ij}) = (1 - r_{ij}/r_c)^2$ for $r_{ij} \leqslant r_c$ and zero for $r_{ij} > r_c$. The effective dissipative force from solid boundaries is extracted from the fluid–solid interactions using the continuum approximation [51]. In particular,
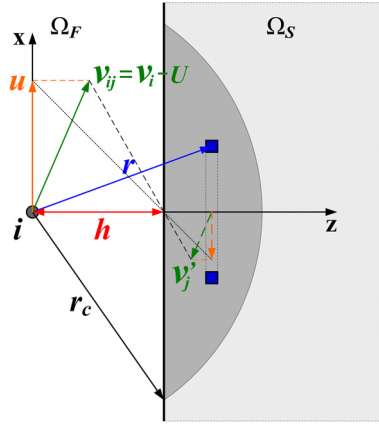
**Fig. 4.** Integration domains for the effective boundary force. $\Omega_F$ and $\Omega_S$ represent the domain of fluid and solid wall, respectively. The number of particles in the infinitesimal ring element is $2\pi \rho g(r)xdxdz$ in which $\rho$ is the average number density and $g(r)$ the radial distribution function. $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{U}$ is the instantaneous velocity difference between particle $i$ and the boundary, while $\mathbf{v}'_j = -\mathbf{v}_{ij} \cdot (z - h)/h$ is an extrapolated velocity for solid particles.

we integrate the force contribution over the part of the cutoff sphere that lies in the solid domain, as shown with the gray area of Fig. 4 where we choose a coordinate system with $x$-axis along the direction of the velocity component of particle $i$ parallel to the wall surface and $z$-axis perpendicular to the wall surface. Let $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{U}$ be the instantaneous velocity difference between particle $i$ with velocity $\mathbf{v}_i$ and the solid particles with velocity $\mathbf{U}$, and $u$ be the parallel component of $\mathbf{v}_{ij}$.

Consider a planar wall surface or a wall surface with the radius of curvature far greater than the cutoff radius $r_c$; if the wall is considered as a rigid body and has uniform velocity $\mathbf{U}$, the instantaneous velocity difference in parallel direction is $u$. Then, the total dissipative force $F_0^D(h)$ on the particle $i$ due to the presence of wall boundary can be evaluated by:

$$
\begin{aligned}
F_0^D(h) &= \int_{z=h}^{r_c} \int_{x=0}^{\sqrt{r_c^2 - z^2}} \int_{\theta=0}^{2\pi} \left( -\gamma \cdot \omega_D(r_{ij})(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij} \cdot \rho \cdot g(r) \cdot dx \cdot x \cdot d\theta \cdot dz \right) \\
&= \int_{z=h}^{r_c} \int_{x=0}^{\sqrt{r_c^2 - z^2}} \left( -\gamma \pi \rho \cdot u \cdot \left(1 - \frac{r}{r_c}\right)^2 \cdot \frac{x^3}{r^2} \cdot g(r) \cdot dx \cdot dz \right) \\
&\xrightarrow{g(r)=1} -\gamma \pi \rho u r_c^3 \left[ \frac{1}{45} - \frac{1}{12}\frac{h}{r_c} - \left(\frac{h}{r_c}\right)^3 \left( \frac{1}{3}\log\left(\frac{h}{r_c}\right) + \frac{2}{9} \right) + \frac{1}{3}\left(\frac{h}{r_c}\right)^4 - \frac{1}{20}\left(\frac{h}{r_c}\right)^5 \right].
\end{aligned} \tag{10}
$$

However, the value of $F_0^D(h)$ is not sufficient to impose the correct no-slip boundary condition on the wall surface. To this end, we assign an extrapolated velocity $\mathbf{v}'_j = -\mathbf{v}_{ij} \cdot (z - h)/h$ to each solid particle so that the wall surface has zero velocity, and hence the instantaneous velocity difference becomes $\tilde{\mathbf{v}}_{ij} = \mathbf{v}_{ij} - \mathbf{v}'_j = \mathbf{v}_{ij} \cdot z/h$, in which the parallel component is $u \cdot z/h$. Then, the corrected dissipative force $F_{\text{cor}}^D(h)$ on the particle $i$ due to the presence of wall boundary is computed by:

$$
\begin{aligned}
F_{\text{cor}}^D(h) &= \int_{z=h}^{r_c} \int_{x=0}^{\sqrt{r_c^2 - z^2}} \int_{\theta=0}^{2\pi} \left( -\gamma \cdot \omega_D(r_{ij})(\mathbf{e}_{ij} \cdot \tilde{\mathbf{v}}_{ij})\mathbf{e}_{ij} \cdot \rho \cdot g(r) \cdot dx \cdot x \cdot d\theta \cdot dz \right) \\
&= \int_{z=h}^{r_c} \int_{x=0}^{\sqrt{r_c^2 - z^2}} \left( -\gamma \pi \rho \cdot \left(\frac{z}{h}u\right) \cdot \left(1 - \frac{r}{r_c}\right)^2 \cdot \frac{x^3}{r^2} \cdot g(r) \cdot dx \cdot dz \right) \\
&\xrightarrow{g(r)=1} -\gamma \pi \rho u r_c^3 \left[ \frac{1}{240}\frac{r_c}{h} - \frac{1}{24}\frac{h}{r_c} - \frac{1}{4}\left(\frac{h}{r_c}\right)^3 \left[ \log\left(\frac{h}{c}\right) + \frac{3}{4} \right] + \frac{4}{15}\left(\frac{h}{r_c}\right)^4 - \frac{1}{24}\left(\frac{h}{r_c}\right)^5 \right].
\end{aligned} \tag{11}
$$

Fig. 5(a) plots the distance-dependent functions of $F_0^D(h)$ and $F_{\text{cor}}^D(h)$, which shows that the correction with extrapolated velocities does not change the value of $F^D(h)$ significantly at large distances (i.e., $h/r_c > 0.5$) but yields bigger dissipative force at small distances (i.e., $h/r_c < 0.5$). Here, we define the ratio of $F_{\text{cor}}^D(h)$ to $F_0^D(h)$ as a correction coefficient $\lambda(h) =$
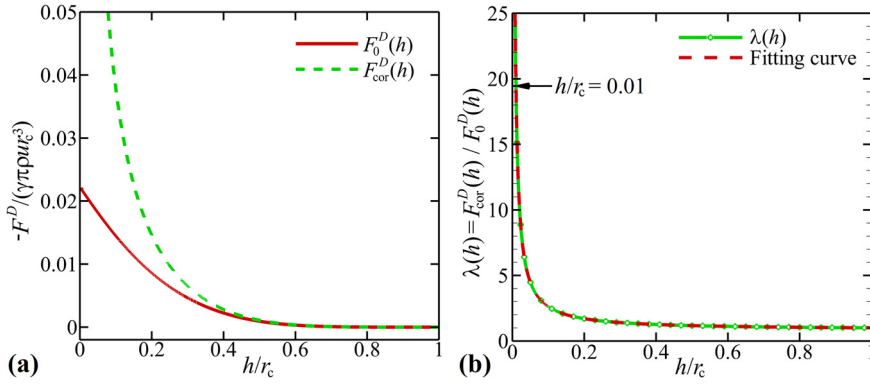
**Fig. 5.** (a) Distance-dependent functions of the effective dissipative force $F_0^D(h)$ and $F_{cor}^D(h)$. (b) Correction coefficient $\lambda(h)$ defined by the ratio of $F_{cor}^D(h)$ to $F_0^D(h)$ and its approximation given by Eq. (12).
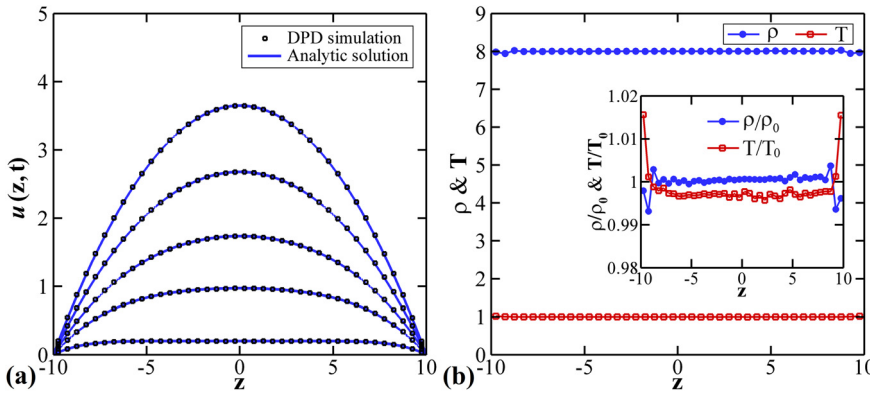


**Fig. 6.** (a) Time evolution of the velocity profile $u(z,t)$ and comparison with analytic solution at $t = 10, 50, 100, 200$ and at steady state, and (b) density $\rho$ and temperature $T$ profiles in a Poiseuille flow. The inset of (b) shows negligible fluctuations of density ($< 1\%$) and temperature ($< 2\%$) in the vicinity of the wall surface. The simulation uses 24000 fluid particles and 4800 frozen particles for two flat walls of thickness 2.0 in a computational domain of $30.0 \times 5.0 \times 24.0$ with $\rho = 8.0, a = 75.0 k_B T / \rho, \sigma = 3.0, k_B T = 1.0$ and $r_c = r_{cw} = 1.0$.

$F_{cor}^D(h)/F_0^D(h)$, which is plotted in Fig. 5(b). In practical implementation, the distance-dependent coefficient $\lambda(h)$ can be approximated by

$$\lambda(h) = \lambda(\varphi^{-1}(\phi) \cdot r_{cw}) \approx \begin{cases} 1 + 0.187\left(\frac{r_c}{h} - 1\right) - 0.093\left(1 - \frac{h}{r_c}\right)^3 & 0.01 \leqslant h/r_c \leqslant 1.0 \,, \\ 19.423 & h/r_c < 0.01 \,. \end{cases} \tag{12}$$

Let the effective dissipative coefficient for liquid–solid interaction be $\gamma_e = \lambda(h) \cdot \gamma$. We note that the formula of Eq. (12) is obtained based on $g(r) = 1$. A more accurate function of $\lambda(h)$ can be derived from Eqs. (10) and (11) using the computed $g(r)$. For easier numerical implementation using Eq. (12) directly without computation of $g(r)$, it is recommended to keep $N_w = 2\pi r_{cw}^3/3 \cdot \rho_w \geqslant 15$, i.e., setting $r_{cw} \geqslant 1.35$ at $\rho_w = 3$ and $r_{cw} \geqslant 1.0$ at $\rho_w = 8$, so that the value of $\phi$ can be evaluated accurately. Then, the dissipative force between liquid particles and solid particles is computed by $\mathbf{F}_{ij}^D = -\gamma_e \cdot \omega_D(r_{ij})(\mathbf{e}_{ij} \cdot \mathbf{v}_{ij})\mathbf{e}_{ij}$, which guarantees the no-slip boundary condition at the wall surface. The corresponding random force is given by $\mathbf{F}_{ij}^R = \sigma_e \cdot \omega_R(r_{ij}) d\tilde{W}_{ij} \mathbf{e}_{ij}$ with $\sigma_e = 2k_B T \gamma_e$ and $d\tilde{W}_{ij}$ being independent increments of the Wiener process to satisfy the FDT [33]. The physical motivation for using a continuum approximation in Eqs. (10) and (11) is the fact that a discrete particle-based system with number density $\rho \to \infty$ converges to a continuum representation. The discretization error is induced by the finite particle density of DPD systems. In the next section, we will verify the validity and the accuracy of the boundary method using the effective dissipative coefficient for liquid–solid interaction.

## 3. Numerical results

In this section, we examine the accuracy of the proposed boundary method for well-known flows such as the plane Poiseuille flow, the plane Couette flow and the Wannier cylindrical flow. Then, a demonstration of flow in a "Brown Pacman" microfluidic device involving very complex boundaries is performed.

Firstly, we test the accuracy of the boundary method on stationary walls by carrying out a DPD simulation of the plane Poiseuille flow, in which a body force field acting in the $x$-direction on a fluid between two flat plates in the $xy$-plane. In this simple case, the Navier–Stokes equations admit the exact solution of the velocity profile given by [52]

$$u(z,t) = \frac{Fd^2}{8\upsilon}\left(1 - \left(\frac{2z}{d}\right)^2\right) - \sum_{n=0}^{\infty} \frac{4(-1)^n Fd^2}{\upsilon\pi^3(2n+1)^3} \cdot \cos\left[\frac{(2n+1)\pi z}{d}\right] \cdot \exp\left[-\frac{(2n+1)^2\pi^2\upsilon t}{d^2}\right], \tag{13}$$

where $d$ is the separation of the plates, $\upsilon$ the kinematic viscosity and $F$ a driving force per unit mass. The parameter set for the Poiseuille flow is $\rho = 8.0$, $k_B T = 1.0$, $a = 75.0 k_B T/\rho$, $\gamma = 4.5$, $\sigma = 3.0$ and $r_c = r_{cw} = 1.0$. The kinematic viscosity of the DPD fluid can be computed by running a periodic Poiseuille flow [53], which gives $\upsilon = 0.275$.

More specifically, the DPD simulation of transient Poiseuille flow is performed in a computational domain of $30.0 \times 5.0 \times 24.0$ in DPD units, which contains 24000 fluid particles and 4800 frozen particles for solid walls with a thickness of 2.0. The system is initialized with stationary fluid and two stationary walls. Periodic boundary condition is applied in $x$- and $y$-directions and no-slip boundary condition in $z$-direction. Then, a body force $g_x = 0.02$ is applied on each DPD particle to drive the fluid, which is equivalent to imposing a pressure drop of $\rho g_x L_x$ on the channel of length $L_x$. To extract the velocity profile from the DPD simulation, we divide the computational domain into 48 bins of width $\Delta = 0.5$ along the $z$-direction. The transient velocity profiles at $t = 10, 50, 100, 200$ and at steady state are plotted in Fig. 6(a), where all local flow properties including particle density and kinetic temperature are obtained by averaging enough sampled data from 100 independent simulations initialized with different random seeds. The first and last bins contain both fluid and solid volumes because of the roughness of the wall surface, as shown in Fig. 2. Considering the flat solid walls are made of randomly distributed particles, the volume of the raised part equals to the volume of the sunk part on average. Therefore, when we compute the density profile, all the fluid particles of $z < 0.5$ are collected into the first bin and the fluid particles of $z > 19.5$ are collected into the last bin. In Fig. 6(a) we observe that the transient velocity profiles are in an excellent agreement with the analytical solution given by Eq. (13), which indicates that the boundary method can provide accurate no-slip boundary condition on the wall surface. Furthermore, Fig. 6(b) shows that our boundary method gives negligible density fluctuation (less than 1%) and temperature fluctuation (less than 2%) in the vicinity of the wall boundary.

The next test case is used to validate the boundary method for moving flat walls. The Couette flow considers a viscous DPD fluid between two parallel plates, one of which is moving relative to the other. To simplify the case, we make the upper wall moving and the lower wall stationary. Similarly to the first test case, the DPD simulation of the Couette flow is performed in a computational domain of $30.0 \times 5.0 \times 24.0$ with periodic boundary conditions in $x$- and $y$-directions and no-slip solid walls in $z$-direction. By solving a one-dimensional Navier–Stokes equation with boundary conditions of $u(0,t) = 0$ and $u(20,t) = 1.0$, an analytical solution for the transient velocity profile $u(z,t)$ can be obtained [54]

$$u(z,t) = \frac{z}{d}U_0 + \frac{2U_0}{\pi}\sum_{n=1}^{\infty}\frac{(-1)^n}{n}\sin\left(\frac{n\pi}{d}z\right)\exp\left(-\frac{n^2\pi^2}{d^2}\nu t\right), \tag{14}$$

where $U_0$ is the velocity of the moving wall, $d$ is the separation between two plates and $\nu$ the kinematic viscosity of the fluid. The computational domain is divided into 48 bins of width $\Delta = 0.5$ along the $z$-direction for obtaining local velocity profiles and fluid properties. Fig. 7(a) shows a comparison between the transient velocity profiles obtained by DPD simulation and the theoretical solution of Eq. (14) at several times and also the steady state solution. The results are in good agreement, which validates the proposed boundary method for imposing the correct no-slip boundary condition for moving walls in the DPD simulation. Furthermore, similarly to the test case of Poiseuille flow, in Fig. 7(b) we observe negligible density fluctuation (less than 1%) and temperature fluctuation (less than 2%) in the vicinity of wall boundary.

In the previous two cases, we have tested the performance of the proposed boundary method for both stationary and moving flat walls. It yields correct no-slip boundary and successfully prevents the liquid particles from penetrating into solid boundaries. The next test case is for curved wall boundaries. We consider the so-called Wannier flow [55] of two eccentric rotating cylinders shown in Fig. 8 for validating the boundary method on curved walls since it involves both concave and convex wall boundaries. For this problem we setup the system with cylinder radii of $R_{outer} = 10$ at center $C_{outer} = \{0, 0, 0\}$ and $R_{inner} = 5.0$ at center $C_{inner} = \{0, -2.5, 0\}$. The outer cylinder is set to rotate with an angular velocity of $\Omega = R_{outer}^{-1} = 0.1$ while the inner cylinder is stationary. Then, the velocity and acceleration on the outer cylinder surface are $\mathbf{U} = \{u, v, w\} = \{-y\Omega, x\Omega, 0\}$ and $\mathbf{a} = \{a_x, a_y, a_z\} = \{-\Omega^2 x, -\Omega^2 y, 0\}$, respectively, which will be used in the predictor–corrector algorithm given by Eq. (4).

To construct a DPD system for the Wannier flow, we first fill a computational domain of $22.4 \times 22.4 \times 10.0$ with 40141 randomly distributed DPD particles. The DPD parameters are set as $\rho = 8.0$, $a = 75.0 k_B T/\rho$, $\sigma = 3.0$, $k_B T = 1.0$ and $r_c = r_{cw} = 1.0$. Then, we relax the system by running a short DPD simulation for 1000 time steps to eliminate the initial configurational energy. Subsequently, the DPD particles with $10.0 \leq R = \sqrt{x^2 + y^2} \leq 11.2$ are defined as outer cylinder, and the DPD particles with $3.8 \leq R = \sqrt{x^2 + y^2} \leq 5.0$ are defined as inner cylinder, while particles with $5 < R = \sqrt{x^2 + y^2} < 10.0$ are fluid particles. All other particles are removed from the system. Finally, the DPD system has 18850 fluid particles and two cylinders of thickness 1.2 consisting of 9048 solid particles. In the simulation, the boundary surface is defined by the isosurface of $\phi = 0.5$. The Reynolds number of the Wannier flow is $Re = U_0 G/\nu = 1.0 \times 2.5/0.275 = 9.09$, in which $U_0$ is
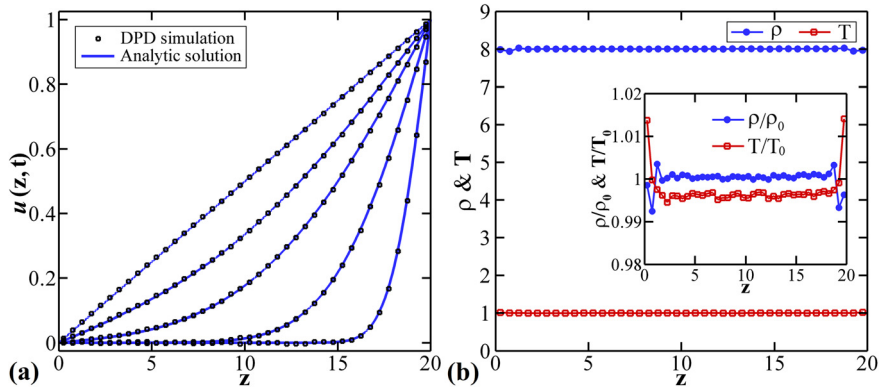
**Fig. 7.** (a) Velocity $u$, (b) density $\rho$ and temperature $T$ profiles and comparison with Navier–Stokes solution in a Couette flow. The inset of (b) shows that the fluctuations of density and temperature in the vicinity of the wall surface are less than 2%. The simulation uses 24000 fluid particles and 4800 solid particles for two flat walls with a thickness of 2.0 in a computational domain of $30.0 \times 5.0 \times 24.0$. DPD parameters are set by $\rho = 8.0, a = 75.0 k_B T/\rho, \sigma = 3.0, k_B T = 1.0$ and $r_c = r_{cw} = 1.0$.
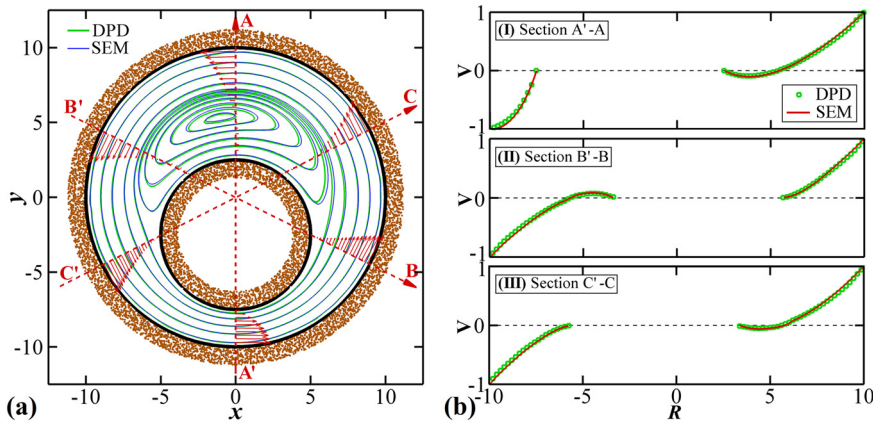


**Fig. 8.** (a) Streamlines of the Wannier flow and (b) velocity profiles of sections A′–A, B′–B and C′–C obtained by the DPD simulation and the spectral element method (SEM). The DPD simulation uses 18850 fluid particles and 9054 solid particles for two cylinders of thickness 1.2 in a computational domain of $22.4 \times 22.4 \times 10.0$. The wall surface is represented by the isosurface of $\phi = 0.5$ in the DPD simulation.

the velocity on the surface of outer cylinder, $G$ the gap between two cylinders and $\nu$ the kinematic viscosity. The DPD fluid has a compressibility of $\kappa^{-1} = 15.40$ and a sound speed of $c = 3.92$ in reduced DPD units, which indicates a Mach number $Ma = U_0/c = 0.255$ for the Wannier flow. We run the DPD simulation for 5 000 time units to obtain a fully developed Wannier flow with a time step $\Delta t = 0.005$. To obtain the local velocity vectors from the DPD simulation, the computational domain is divided into square cells with side length 0.25 in $xy$-plane. The flow field shown in Fig. 8(a) is obtained by averaging particle velocity in each cell over 10 000 time units. We note that the accuracy of the flow field obtained from DPD simulations is determined by the number of samples in each cell (by performing space–time average and ensemble average). In the present study, we follow the DPD parameters used in the Poiseuille flow with a number density of 8.0. If a different number density rather than 8.0 is employed, one needs to carefully parameterize the DPD system to have the same dimensionless numbers, i.e., $Re$ and $Ma$, to generate the same results. Moreover, for a DPD system with lower number densities, more space–time average or ensemble average should be performed to ensure the same quality of the flow field.

We compare the streamlines of the Wannier flow in Fig. 8(a) and the velocity profiles of sections A′–A, B′–B and C′–C in Fig. 8(b) obtained by the DPD simulation with the result obtained by the spectral element method (SEM) for an incompressible flow at the same Reynolds number of $Re = 9.09$. Results show that the DPD simulation is in very good agreement with the solution of SEM, where small deviation of streamlines may be induced by the compressibility of the DPD fluid. As long as we run the DPD simulation, we do not observe wall penetration, which indicates that the proposed boundary method can be safely applied to problems involving curved boundaries. The random surface roughness is generated by the random-distributed particle representation, which may change the hydrodynamics locally. However, in our benchmark tests, as far as we have tested, we saw no significant changes on the averaged velocity fields.

For wall boundaries that can be defined analytically, there is no difference between the local detection method and other boundary methods for DPD [37,49,50]. The advantage of this method is on-the-fly detection of boundaries (at a cheap computational cost), so that it can be applied to not only simple geometries that can be analytically defined, but
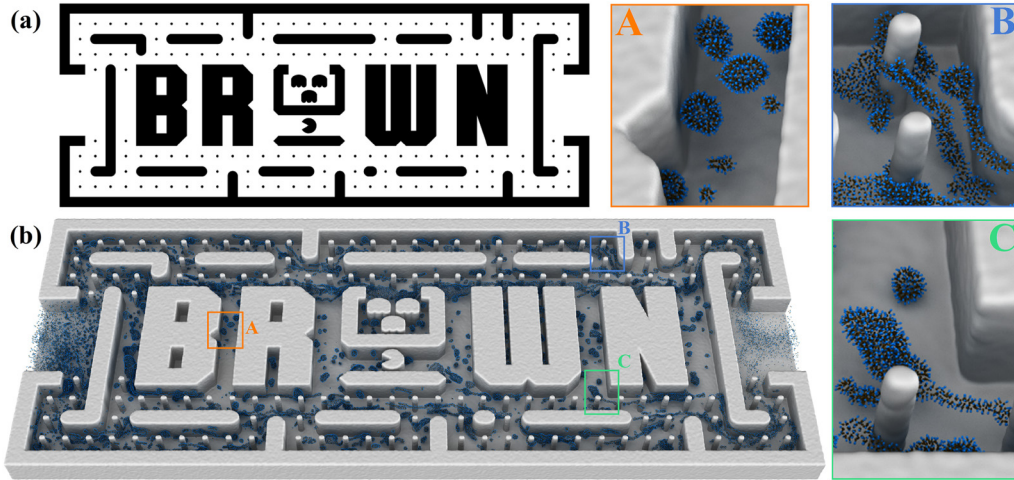
**Fig. 9.** (a) The vector graphical image used for generating the DPD system of a "Brown Pacman" microfluidic device. (b) Visualization of the surfactant solution flowing through the "Brown Pacman" channel (see also Supporting Information for the movie). A, B and C are three zoom-in views of (b).

also complicate geometries that can hardly be described mathematically. To further demonstrate the capability of the local detection boundary method in realistic application scenarios, we construct a "Brown Pacman" microfluidic device and carry out a simulation of a surfactant solution flowing through the microfluidic channel with complex geometry [56]. The system is set up by mapping a vector graphics image of the desired channel geometry, as shown in Fig. 9(a), onto a simulation box of size $600 \times 230 \times 24$ reduced units. DPD particles representing the channel wall are then placed randomly within regions with brightness $< 50\%$, while 6 341 124 solvent particles and 300 000 surfactant particles with a volume concentration of 4.52% are randomly placed in regions with brightness $> 50\%$. The system comprises of a total of 13 248 000 DPD particles with a number density $\rho = 4$ and the simulation is performed using the $_{USER}$MESO GPU-accelerated DPD package [57]. Each surfactant molecule has one hydrophilic bead (H) and one hydrophobic bead (T) connected by a harmonic bond with potential $E_b(r) = K(r - r_0)^2$, where $K$ is the spring force constant, and $r$, $r_0$ the instantaneous and equilibrium bond length. The two beads representation $(H_1T_1)$ is the simplest model for surfactants, phospholipids and block copolymers used in chemical engineering [58,59]. We note that longer chains $(H_mT_n, \ m + n > 2)$ for copolymers should be used if molecule structures are important in application problems. A cutoff distance $r_c = 1.0$ is used for the pairwise interaction and $r_{cw} = 1.0$ for the local detection method. The wall surface is represented by the isosurface of $\phi = 0.5$, but the isosurface of $\phi = 0.5$ can deviate from the real curved surface (especially at sharp corners), and their deviation increases as the curvature increases. Since the local detection method imposes no-slip boundary condition on the isosurface of $\phi = 0.5$ (not real physical wall surface), for both concave and convex geometries, it yields a curvature-dependent slip for velocity on the physical boundaries. Quantitative details about the deviation of the isosurface of $\phi = 0.5$ from real physical boundaries are included in Appendix A. The interaction matrix between the surfactant, solvent and wall particles is given in Table 1. The hydrophobic/hydrophilic properties of the surfactant molecule is determined by the Flory–Huggins $\chi$-parameter [32], i.e., negative $\chi$-parameter for hydrophilic beads and positive $\chi$-parameter for hydrophobic beads. A lateral pressure gradient, $-\partial p/\partial x = c(v_x - v_x^0)$, where $c = 0.25$ and $v_x^0 = 4$, is applied at the inlet of the channel to drive the flow. The system is first optimized using a short run of DPD simulation. A time step size of $\Delta t = 0.01$ is then used to simulate the system for $1 \times 10^6$ time steps.

In Fig. 9(b), we observe rich phenomena of surfactant dynamically assembling and disassembling following the flow (see also Supporting Information for the movie). Three local zoom-in views of Fig. 9(b) are shown in Fig. 9A, B and C. More specifically, zone A is located between the walls of "B" and "R", where the flow field is almost stationary. Consequently, the surfactant molecules in a shear-free solution self-assemble into small spherical aggregates, as shown in Fig. 9A. However, Fig. 9B shows that the surfactant molecules form elongated wormlike micelles under strong shear flow. We observe that these wormlike micelles flow around the small cylinders without wall penetration. Zone C is located at a transition area from a nearly stationary flow to a shear flow, where a shear-induced phase transition from spherical micelles to elongated wormlike micelles is shown in Fig. 9C. Since the boundary geometry is computed on-the-fly, the proposed local detection method takes care of imposing no-slip boundary conditions and preventing wall penetrations automatically, even for such a complex microfluidic device. This may be very valuable for many realistic applications.

## 4. Summary and discussions

A local detection method tackling the challenges induced by arbitrarily shaped boundaries and complex geometries in dissipative particle dynamics (DPD) simulations has been proposed. By computing a boundary volume fraction (BVF) for each fluid particle, the solid boundary is detected on-the-fly by the fluid particles according to local particle configuration.

**Table 1**
Repulsive force constants $a_{ij}$ for microfluidic channel.

|          | H    | T    | Solvent | Wall |
|----------|------|------|---------|------|
| H        | 45   | 75   | 37.5    | 150  |
| T        | 75   | 37.5 | 150     | 150  |
| Solvent  | 37.5 | 150  | 37.5    | 37.5 |
| Wall     | 150  | 150  | 37.5    | 37.5 |

At a small extra computational cost, the fluid particles become autonomous to find the wall surface and to infer the wall penetration based on the coordinates of their neighboring particles. A predictor–corrector algorithm was employed to prevent the fluid particles from penetrating into the wall boundaries, and the effective dissipative coefficients for liquid–solid interactions were used to impose no-slip boundary condition on the wall surface.

We employed randomly distributed particles to represent walls to allow easiness and generality for construction of DPD systems involving arbitrary-shape boundaries. Theoretically, to the limit of the continuum approximation, the wall surface can be accurately represented by the isosurface of BVF $\phi = 0.5$ when the solid particles are dense enough. However, in practical implementations, the random distribution of discrete particles with finite number density will introduce surface roughness of wall boundaries, which comes from the estimation error of BVF based on the discrete particles. We demonstrated that the magnitude of roughness associated with the randomness of particle distribution is monotonically controllable by tuning the cutoff radius for computing BVF and the number density of DPD particles. Since any natural solid wall contains more or less chemical/physical heterogeneities on the surface, especially at the mesoscopic scale, such heterogeneity can be modeled qualitatively by the roughness on walls in the DPD systems. In this respect, the proposed boundary method provides a convenient way to generate various sizes of the roughness for representation of different degrees of the heterogeneity and to introduce curvature-dependent slip for hydrodynamics as discussed in the Appendix A.

The transient Poiseuille and Couette flows as well as the Wannier flow were used as benchmark tests for verifying the proposed arbitrary boundary method. The results showed that the proposed boundary method imposes the correct no-slip boundary condition for both stationary and moving walls in the DPD simulation, and yields negligible density fluctuation (less than 1%) and temperature fluctuation (less than 2%) in the vicinity of wall surface. The local detection method can be applied to complex fluids straightforward without any additional efforts. To further demonstrate the capability of the presented arbitrary boundary method in realistic application scenarios, a "Brown Pacman" microfluidic device with complex geometry was constructed directly from a vector graphics image and a DPD simulation of surfactant solution flowing through this complex microfluidic device was carried out. The validity of this boundary method was confirmed by examining the rich dynamics of surfactant micelles flowing around the small cylinders without wall penetration.

The local detection method was implemented as an extension to the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [60] and the GPU-accelerated DPD package $_{USER}$MESO [57,61]. Since this method only uses local information of neighboring particles for computing the value of BVF and satisfies designed boundary conditions on-the-fly, it provides a practical and efficient way to deal with complex geometries and impose the no-slip boundary condition on wall surface in DPD simulations. With the local detection method, it is no longer necessary to mathematically define the boundary geometry for DPD simulations, which enables us to construct DPD systems directly from experimental CT images or computer-aided designs/drawings. Moreover, this method is not only valuable for stationary arbitrary-shape boundaries, but also for the moving boundaries and deformable boundaries.

The DPD method is a bottom-up approach starting from microscopic dynamics, which was designed for investigating mesoscopic dynamics of complex fluids [32]. In the continuum limit, the hydrodynamic equations of a DPD system can recover the continuity and Navier–Stokes equations [62]. However, for simple fluids, whose dynamics can be correctly described by macroscopic partial differential equations, DPD cannot perform better than top–down methods derived from continuum equations, such as the fluctuating hydrodynamics [63] and the smoothed DPD method [10,64].

Although we presented here that the surface roughness is controllable by varying the number of neighboring particles, this boundary method cannot accurately capture large curvatures of wall boundary where the radius of curvature is too small to be identified from the surface roughness. To this end, higher resolution of DPD system is required to represent the large curvature properly so that the size of random surface roughness is much smaller than the radius of curvature.
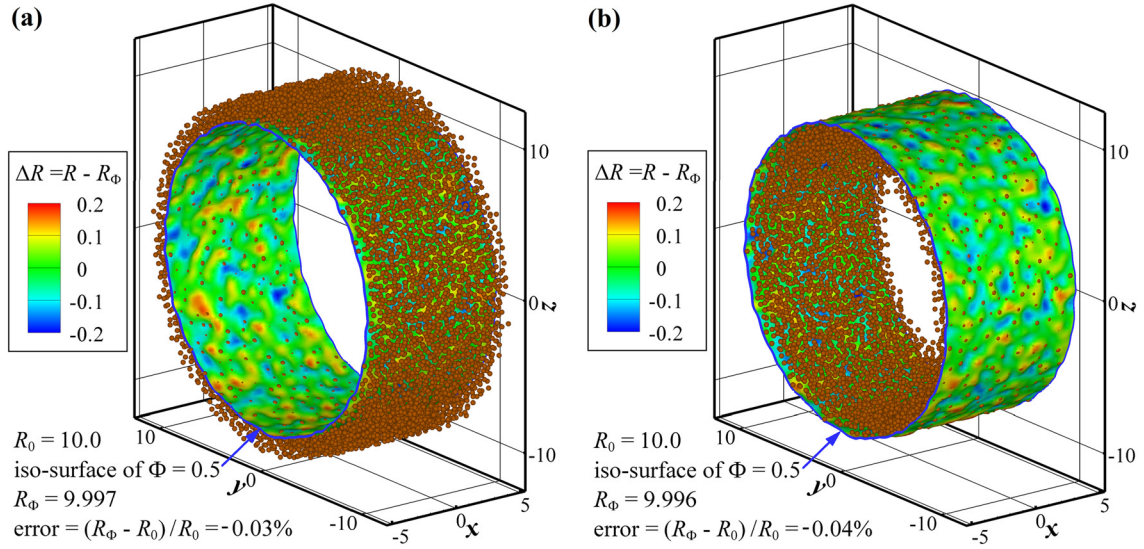
### Acknowledgements

**Fig. 10.** Performance accuracy of representing curved wall boundaries, including (a) concave surface and (b) convex surface, by the isosurface of $\phi = 0.5$. The expected radii of cylinders are $R_0 = 10.0$, the isosurface of $\phi = 0.5$ gives $R_\Phi = 9.997$ for the concave cylinder and $R_\Phi = 9.996$ for the convex cylinder. Spherules represent randomly distributed solid particles constituting cylinders.
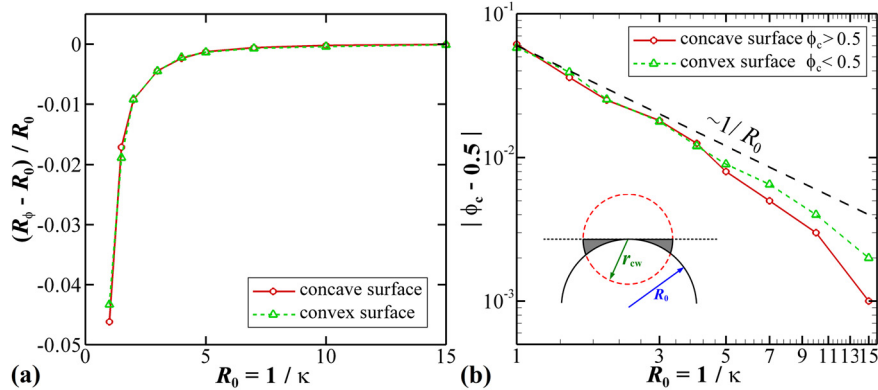


**Fig. 11.** (a) Relative error $(R_\Phi - R_0)/R_0$ vs radius of curvature of the curved wall boundaries $R_0 = 1/\kappa$ in which $\kappa$ is the curvature. (b) Corrected magnitude of boundary friction function $\phi_c$ that accurately represents the expected concave and convex wall boundaries.

## Appendix A. Error analysis for curved surfaces

Representing the wall boundary by the isosurface of $\phi = 0.5$ is accurate for flat wall boundaries, as indicated by Eq. (5). However, for curved boundaries, the isosurface of $\phi = 0.5$ will deviate from the designed boundary surface. To analyze the error and the performance of representing curved wall boundaries with the isosurface of $\phi = 0.5$, we construct a series of concave and convex cylinders to quantify the error induced by the curvature of wall boundaries. Fig. 10 shows the geometry of concave and convex cylinders. The expected radii of both cylinders are $R_0 = 10.0$. Let $\rho_w = 8.0$ be the particle number density, the average separation between particles is $\delta = \rho_w^{-1/3} = 0.5$. We fill a computational domain of $10.0 \times 25.0 \times 25.0$ with 50000 randomly distributed DPD particles. Then, we relax the system by running a short DPD simulation to eliminate initial randomicity. For the concave cylinder, the DPD particles with $R = \sqrt{y^2 + z^2}$ ranging from 10.0 to 12.0 are defined as solid particles, as shown in Fig. 10(a). Similarly, the convex cylinder is made up by the discrete DPD particle with $R = \sqrt{y^2 + z^2}$ ranging from 8.0 to 10.0.

By defining $r_{cw} = 1.0$, the value of $\varphi(\mathbf{r})$ can be easily computed using Eq. (2). Fig. 10 shows the performance of representing concave and convex boundaries by the isosurfaces of $\phi = 0.5$. The designed radii of both two cylinders are $R_0 = 10.0$, while the iso-surface of $\phi = 0.5$ gives $R_\Phi = 9.997$ for the concave cylinder and $R_\Phi = 9.996$ for the convex cylinder. Although the isosurfaces of $\phi = 0.5$ slightly deviate from the expected cylinder radius, the relative error $(R_\Phi - R_0)/R_0$ are negligible for $R_0 = 10.0$, i.e., less than 0.05%.

Let $\kappa$ be the curvature of the cylinder surface, as the radius of curvature $R_0 = 1/\kappa$ decreases, the relative error $(R_\Phi - R_0)/R_0$ increases, as plotted in Fig. 11(a). We observe that the relative error $(R_\Phi - R_0)/R_0$ is about 4.5% for $R_0 = 1.0$,

however, it becomes less than 1% for $R_0 \geq 2.0$. Consequently, using the isosurface of $\phi = 0.5$ to represent the curved boundaries does not induce significant error for small curvatures, i.e., $\kappa = R_0^{-1} \leq 0.5$.

Theoretically, a corrected magnitude of the boundary friction function $\phi_c$ rather than 0.5 can be defined to accurately represent the curved boundaries. Using the continuum approximation, the extra volume for curved surfaces different from a flat surface is $\pi r_{cw}^4 / 4R_0$, as shown by the dark domain in the inset of Fig. 11(b). Then, the corrected boundary friction functions are $\phi_c = 0.5 + 3r_{cw}/16R_0$ for concave surfaces and $\phi_c = 0.5 - 3r_{cw}/16R_0$ for convex surfaces. Therefore, the difference between $\phi_c$ and 0.5 decreases as the radius of curvature $R_0 = 1/\kappa$ increases, which is given by

$$|\phi_c - 0.5| = \frac{\pi r_{cw}^4 / 4R_0}{4\pi r_{cw}^3 / 3} = \frac{3r_{cw}}{16R_0} . \tag{A.1}$$

The formula of Eq. (A.1) is obtained based on the continuum approximation. For the wall consisting of discrete DPD particles, we observe in Fig. 11(b) that the decay of $|\phi_c - 0.5|$ is slightly faster than $R_0^{-1}$. This is because the distribution of DPD particle is initially regularized by performing a short simulation.

It is worth noting that employing the isosurface of $\phi = 0.5$ to represent the curved boundaries could introduce curvature-dependent slip for hydrodynamics, which would be practically useful for some applications where partial slips on curved wall surfaces are expected. As described in Fig. 11(a), the isosurface of $\phi = 0.5$ deviates from the real curved surface and the deviation increases as the curvature increases. Since the local-detection boundary method imposes no-slip boundary condition on the isosurface of $\phi = 0.5$, for both concave and convex geometries, a partial slip for velocity will be observed on physical boundaries, where the slip length depends on the curvatures of the boundary as shown in Fig. 11(a).

## Appendix B. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jcp.2017.11.014.

## References

[1] W.G. Noid, Perspective: coarse-grained models for biomolecular systems, J. Chem. Phys. 139 (9) (2013) 090901.
[2] E. Brini, E.A. Algaer, P. Ganguly, C.L. Li, F. Rodriguez-Ropero, N.F.A. van der Vegt, Systematic coarse-graining methods for soft matter simulations – a review, Soft Matter 9 (7) (2013) 2108–2119.
[3] S. Yip, M.P. Short, Multiscale materials modelling at the mesoscale, Nat. Mater. 12 (9) (2013) 774–777.
[4] Z.G. Mills, W.B. Mao, A. Alexeev, Mesoscale modeling: solving complex flows in biology and biotechnology, Trends Biotechnol. 31 (7) (2013) 426–434.
[5] M.G. Saunders, G.A. Voth, Coarse-graining methods for computational biology, Annu. Rev. Biophys. 42 (2013) 73–93.
[6] Z. Li, Y.-H. Tang, X.J. Li, G.E. Karniadakis, Mesoscale modeling of phase transition dynamics of thermoresponsive polymers, Chem. Commun. 51 (55) (2015) 11038–11040.
[7] M. Lisal, Z. Limpouchova, K. Prochazka, The self-assembly of copolymers with one hydrophobic and one polyelectrolyte block in aqueous media: a dissipative particle dynamics study, Phys. Chem. Chem. Phys. 18 (24) (2016) 16127–16136.
[8] H. Li, V. Ha, G. Lykotrafitis, Modeling sickle hemoglobin fibers as one chain of coarse-grained particles, J. Biomech. 45 (11) (2012) 1947–1951.
[9] D.Y. Pan, J.X. Hu, X.M. Shao, Lees–Edwards boundary condition for simulation of polymer suspension with dissipative particle dynamics method, Mol. Simul. 42 (4) (2016) 328–336.
[10] X. Bian, S. Litvinov, R. Qian, M. Ellero, N.A. Adams, Multiscale modeling of particle in suspension with smoothed dissipative particle dynamics, Phys. Fluids 24 (1) (2012) 012002.
[11] R.G. Winkler, D.A. Fedosov, G. Gompper, Dynamical and rheological properties of soft colloid suspensions, Curr. Opin. Colloid Interface Sci. 19 (6) (2014) 594–610.
[12] D.S. Bolintineanu, G.S. Grest, J.B. Lechman, F. Pierce, S.J. Plimpton, P.R. Schunk, Particle dynamics modeling methods for colloid suspensions, Comput. Part. Mech. 1 (3) (2014) 321–356.
[13] Z. Li, G.H. Hu, Z.L. Wang, Y.B. Ma, Z.W. Zhou, Three dimensional flow structures in a moving droplet on substrate: a dissipative particle dynamics study, Phys. Fluids 25 (7) (2013) 072103.
[14] Y.X. Wang, S. Chen, Numerical study on droplet sliding across micropillars, Langmuir 31 (16) (2015) 4673–4677.
[15] K. Yang, Z.L. Guo, Lattice Boltzmann study of wettability alteration in the displacement of nanoparticle-filled binary fluids, Comput. Fluids 124 (2016) 157–169.
[16] T. Ye, N. Phan-Thien, B.C. Khoo, C.T. Lim, Dissipative particle dynamics simulations of deformation and aggregation of healthy and diseased red blood cells in a tube flow, Phys. Fluids 26 (11) (2014) 111902.
[17] I.V. Pivkin, Z.L. Peng, G.E. Karniadakis, P.A. Buffet, M. Dao, S. Suresh, Biomechanics of red blood cells in human spleen and consequences for physiology and disease, Proc. Natl. Acad. Sci. USA 113 (28) (2016) 7804–7809.
[18] H.Y. Chang, X.J. Li, H. Li, G.E. Karniadakis, MD/DPD multiscale framework for predicting morphology and stresses of red blood cells in health and disease, PLoS Comput. Biol. 12 (10) (2016) e1005173.
[19] H. Lei, D.A. Fedosov, B. Caswell, G.E. Karniadakis, Blood flow in small tubes: quantifying the transition to the non-continuum regime, J. Fluid Mech. 722 (2013) 214–239.
[20] E. Henry, S.H. Holm, Z.M. Zhang, J.P. Beech, J.O. Tegenfeldt, D.A. Fedosov, G. Gompper, Sorting cells by their dynamical properties, Sci. Rep. 6 (2016) 34375.
[21] X.J. Li, E. Du, H. Lei, Y.-H. Tang, M. Dao, S. Suresh, G.E. Karniadakis, Patient-specific blood rheology in sickle-cell anaemia, Interface Focus 6 (1) (2016) 20150065.
[22] X.J. Li, B. Caswell, G.E. Karniadakis, Effect of chain chirality on the self-assembly of sickle hemoglobin, Biophys. J. 103 (6) (2012) 1130–1140.
[23] J. Zavadlav, M.N. Melo, S.J. Marrink, M. Praprotnik, Adaptive resolution simulation of an atomistic protein in MARTINI water, J. Chem. Phys. 140 (5) (2014) 054114.
[24] H. Li, G. Lykotrafitis, Two-component coarse-grained molecular-dynamics model for the human erythrocyte membrane, Biophys. J. 102 (1) (2012) 75–84.

[25] Y.-H. Tang, Z. Li, X.J. Li, M.G. Deng, G.E. Karniadakis, Non-equilibrium dynamics of vesicles and micelles by self-assembly of block copolymers with double thermoresponsivity, Macromolecules 49 (7) (2016) 2895–2903.
[26] P. Hoogerbrugge, J. Koelman, Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics, Europhys. Lett. 19 (3) (1992) 155–160.
[27] D.A. Fedosov, H. Noguchi, G. Gompper, Multiscale modeling of blood flow: from single cells to blood rheology, Biomech. Model. Mechanobiol. 13 (2) (2014) 239–258.
[28] M.B. Liu, G.R. Liu, L.W. Zhou, J.Z. Chang, Dissipative particle dynamics (DPD): an overview and recent developments, Arch. Comput. Methods Eng. 22 (4) (2015) 529–556.
[29] P. Español, P.B. Warren, Perspective: dissipative particle dynamics, J. Chem. Phys. 146 (15) (2017) 150901.
[30] Z. Li, X. Bian, B. Caswell, G.E. Karniadakis, Construction of dissipative particle dynamics models for complex fluids via the Mori–Zwanzig formulation, Soft Matter 10 (43) (2014) 8659–8672.
[31] Z. Li, X. Bian, X.T. Li, G.E. Karniadakis, Incorporation of memory effects in coarse-grained modeling via the Mori–Zwanzig formalism, J. Chem. Phys. 143 (24) (2015) 243128.
[32] R.D. Groot, P.B. Warren, Dissipative particle dynamics: bridging the gap between atomistic and mesoscopic simulation, J. Chem. Phys. 107 (11) (1997) 4423–4435.
[33] P. Español, P. Warren, Statistical mechanics of dissipative particle dynamics, Europhys. Lett. 30 (4) (1995) 191–196.
[34] R. Kubo, The fluctuation–dissipation theorem, Rep. Prog. Phys. 29 (1) (1966) 255–284.
[35] M.P. Allen, D.J. Tildesley, Computer Simulation of Liquids, Clarendon Press, New York, 1989.
[36] S. Adami, X.Y. Hu, N.A. Adams, A generalized wall boundary condition for smoothed particle hydrodynamics, J. Comput. Phys. 231 (21) (2012) 7057–7075.
[37] I.V. Pivkin, G.E. Karniadakis, A new method to impose no-slip boundary conditions in dissipative particle dynamics, J. Comput. Phys. 207 (1) (2005) 114–128.
[38] P.J. Colmenares, R. Rousse, Effective boundary forces for several geometries in dissipative particle dynamics, Physica A 367 (2006) 93–105.
[39] A.M. Altenhoff, J.H. Walther, P. Koumoutsakos, A stochastic boundary forcing for dissipative particle dynamics, J. Comput. Phys. 225 (1) (2007) 1125–1136.
[40] M. Revenga, I. Zúñiga, P. Español, Boundary conditions in dissipative particle dynamics, Comput. Phys. Commun. 121 (122) (1999) 309–311.
[41] G. Liu, M. Liu, Smoothed Particle Hydrodynamics: A Meshfree Particle Method, World Scientific, 2003.
[42] J.J. Monaghan, Smoothed particle hydrodynamics, Rep. Prog. Phys. 68 (8) (2005) 1703–1759.
[43] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, Astron. J. 82 (12) (1977) 1013–1024.
[44] I.V. Pivkin, G.E. Karniadakis, Controlling density fluctuations in wall-bounded dissipative particle dynamics systems, Phys. Rev. Lett. 96 (20) (2006) 206001.
[45] S. Litvinov, M. Ellero, X.Y. Hu, N.A. Adams, Particle-layering effect in wall-bounded dissipative particle dynamics, Phys. Rev. E 82 (6) (2010) 066704.
[46] N.A. Gatsonis, R. Potami, J. Yang, A smooth dissipative particle dynamics method for domains with arbitrary-geometry solid boundaries, J. Comput. Phys. 256 (2014) 441–464.
[47] D.P. Kroese, T. Taimre, Z.I. Botev, Handbook of Monte Carlo Methods, John Wiley & Sons, Inc., 2011.
[48] D. Duong-Hong, N. Phan-Thien, X. Fan, An implementation of no-slip boundary conditions in DPD, Comput. Mech. 35 (1) (2004) 24–29.
[49] H. Lei, D.A. Fedosov, G.E. Karniadakis, Time-dependent and outflow boundary conditions for dissipative particle dynamics, J. Comput. Phys. 230 (10) (2011) 3765–3779.
[50] S.K. Ranjith, B.S.V. Patnaik, S. Vedantam, No-slip boundary condition in finite-size dissipative particle dynamics, J. Comput. Phys. 232 (1) (2013) 174–188.
[51] Z. Li, Y.-H. Tang, H. Lei, B. Caswell, G.E. Karniadakis, Energy-conserving dissipative particle dynamics with temperature-dependent properties, J. Comput. Phys. 265 (2014) 113–127.
[52] L.D.G. Sigalotti, J. Klapp, E. Sira, Y. Meleán, A. Hasmy, SPH simulations of time-dependent Poiseuille flow at low Reynolds numbers, J. Comput. Phys. 191 (2) (2003) 622–638.
[53] J.A. Backer, C.P. Lowe, H.C.J. Hoefsloot, P.D. Iedema, Poiseuille flow to measure the viscosity of particle model fluids, J. Chem. Phys. 122 (15) (2005) 154503.
[54] Z. Li, A. Yazdani, A. Tartakovsky, G.E. Karniadakis, Transport dissipative particle dynamics model for mesoscopic advection–diffusion–reaction problems, J. Chem. Phys. 143 (1) (2015) 014101.
[55] G.H. Wannier, A contribution to the hydrodynamics of lubrication, Q. Appl. Math. 8 (1) (1950) 1–32.
[56] M. Li, M. Humayun, J.A. Kozinski, D.K. Hwang, Functional polymer sheet patterning using microfluidics, Langmuir 30 (28) (2014) 8637–8644.
[57] Y.-H. Tang, G.E. Karniadakis, Accelerating dissipative particle dynamics simulations on GPUs: algorithms, numerics and applications, Comput. Phys. Commun. 185 (11) (2014) 2809–2822.
[58] F. Alvarez, E.A. Flores, L.V. Castro, J.G. Hernandez, A. Lopez, F. Vazquez, Dissipative particle dynamics (DPD) study of crude oil–water emulsions in the presence of a functionalized co-polymer, Energy Fuels 25 (2011) 562–567.
[59] C. Singh, Y. Hu, B.P. Khanal, E.R. Zubarev, F. Stellacci, S.C. Glotzer, Striped nanowires and nanorods from mixed SAMS, Nanoscale 3 (8) (2011) 3244–3250.
[60] S. Plimpton, Fast parallel algorithms for short-range molecular-dynamics, J. Comput. Phys. 117 (1) (1995) 1–19.
[61] A.L. Blumers, Y.-H. Tang, Z. Li, X. Li, G.E. Karniadakis, GPU-accelerated red blood cells simulations with transport dissipative particle dynamics, Comput. Phys. Commun. 217 (2017) 171–179.
[62] C.A. Marsh, G. Backx, M.H. Ernst, Static and dynamic properties of dissipative particle dynamics, Phys. Rev. E 56 (2) (1997) 1676–1691.
[63] Y.H. Wang, J.K. Sigurdsson, E. Brandt, P.J. Atzberger, Dynamic implicit-solvent coarse-grained models of lipid bilayer membranes: fluctuating hydrodynamics thermostat, Phys. Rev. E 88 (2) (2013) 023301.
[64] P. Español, M. Revenga, Smoothed dissipative particle dynamics, Phys. Rev. E 67 (2003) 026705.