CrossMark

ORIGINAL PAPER

# Dissipative Particle Dynamics (DPD): An Overview and Recent Developments

**M. B. Liu · G. R. Liu · L. W. Zhou · J. Z. Chang**

**Abstract** Dissipative particle dynamics (DPD) is a mesoscale particle method that bridges the gap between microscopic and macroscopic simulations. It can be regarded as a coarse-grained molecular dynamics method suitable for larger time and length scales. It has been successfully applied to different areas of interests, especially in modeling the hydrodynamic behavior of complex fluids in mesoscale. This paper presents an overview on DPD including the methodology, formulation, implementation procedure and some related numerical aspects. The paper also reviews the major applications of the DPD method, especially in modeling (1) micro drop dynamics, (2) multiphase flows in microchannels and fracture networks, (3) movement and suspension of macromolecules in micro channels and (4) movement and deformation of single cells. The paper ends with some concluding remarks summarizing the major features and future possible development of this unique mesoscale modeling technique.

**Keywords** Dissipative particle dynamics (DPD) · Meshfree method · Particle method · Coarse-grained method · Mesoscale · Multiscale

M. B. Liu (✉) · L. W. Zhou
Institute of Mechanics, Chinese Academy of Sciences, # 15 Bei Si Huan Xi Road, Haidian District, Beijing 100190, China
e-mail: liumoubin@gmail.com; liumoubin@imech.ac.cn

G. R. Liu
Aerospace Systems, University of Cincinnati, Cincinnati, OH 45221-0070, USA

J. Z. Chang
School of Mechatronic Engineering, North University of China, Taiyuan 030051, China

## 1 Introduction

By integrating mechanical elements, sensors, actuators, and electronic components using microfabrication technology, microelectromechanical systems (MEMS) can be designed to be fast in response, capable of achieving high spatial resolution, and cost-effective in mass production, due to the batch micromachining techniques [1,2]. Since its emergence, MEMS technology has found many important applications to chemical, biological and medical sciences and engineering. For example, MEMS for biomedical and biological applications (usually referred to as BioMEMS) are capable of delivering, processing and analyzing biochemical materials in a wide range of problems, such as disease diagnosis, clinical assays, drug screening and delivery, and even gene searching and sequencing. BioMEMS usually are more efficient and more effective than traditional biomedical and biological techniques. It is also observed that most of the BioMEMS devices use features of microfluidics. Therefore, characterization of fluid flows in MEMS devices has increasingly becoming a very important topic since the fluidic behavior in MEMS is very different from what observed in daily life experienced at macroscales [3,4].

One typical feature of fluid flows in MEMS devices is the size effects. For example, the delivery of drugs is usually conducted by the movement and suspension of macromolecules in micro-channels, where the size of the drug agents (usually DNA molecules) and the size of the micro-channel are important to understand the effects of the macromolecular conformation. If the Knudsen number, $Kn$, defined here as the ratio of the macromolecular length to the characteristic length of flow field, is much smaller than unity (e.g., $Kn \ll 1$), the movement and suspension of macromolecules (in macro channels) can be as regarded a continuum flow. If the Knudsen number is around (or even bigger than) unity,

the movement and suspension of macromolecules (in micro-channels) may not be regarded as a continuum flow. The suspension of DNA in micro-channel is exactly the case with $Kn \simeq O(1)$, as the length of a typical DNA molecule is usually in the same order as the size of a typical micro-channel. For example, the size of a typical micro-channel is about 9–40 µm [5], and the uncoiled length of a λ-DNA is about 22 µm [6] to 33 µm [7,8]. Hence the standard rheological models developed from continuum assumptions for continuum applications may be misleading in describing such flows with $Kn \simeq O(1)$. Also as the molecular dynamics simulation is not feasible for modeling such flows as it is yet restricted from practical applications due to the extremely small time scales (nanoseconds) and length scales (nanometers). Therefore the development of numerical methods at mesoscale is required for effective modeling of microfluidics behaviors.

In general, mesoscale denotes the (length and time) scale larger than atomic scale, but smaller than macroscale. The definition of mesoscale is not rigid and can be different in computational material science, computational physics, computational biology, chemistry, and computational mechanics. For example, in computational material science and computational mechanics, mesoscale usually involves a characteristic length ranging from $10^{-7}$ to $10^{-4}$ m and a characteristic time ranging from $10^{-9}$ to $10^{-3}$ s. This overlaps the microscale (a characteristic length ranging from $10^{-8}$ to $10^{-6}$ m and a characteristic time ranging from $10^{-11}$ to $10^{-8}$ s) and macroscale (a characteristic length bigger than $10^{-4}$ m and a characteristic time bigger than $10^{-3}$ s) (Fig. 1).

For problems at different scales, different computational models should be correspondingly used [9–11]. For macroscale problems, computational models such as the finite element method (FEM) [12–14], smoothed finite element method (S-FEM) [15] finite difference method (FDM) or finite volume method (FVM) [16–19], as well as mesh-free methods [20–24] can be used. These macroscale computational models usually involve constitutive relations to solve a system of partial different equations established based on the assumption of continuum of media. When the length scale gradually reduces, the constitutive relations based on continuum assumptions may no longer be valid. For nano and microscale problems, the atomistic models such as the classic molecular dynamics (MD) [25–27], Ab initio MD [28,29], and monte carlo (MC) [30–32] can be used. The atomistic models provide a fundamental way of obtaining a better understanding of the behavior of fluid flow in micro-channels. However, due to the very small length and time scales associated with these methods, they are computationally expensive, even for modern supercomputers, and they cannot be applied to many important scientific and practical problems.

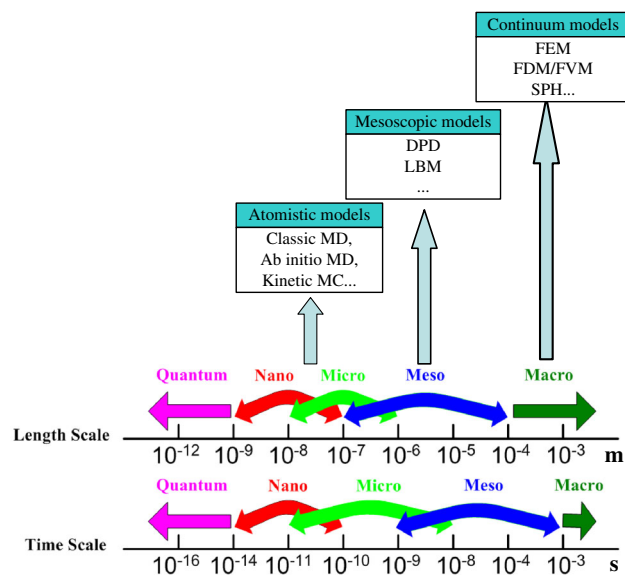Some mesoscopic or coarse-grained simulation methods have been developed during the last two decades. The



**Fig. 1** Different length and time scales and corresponding computational methods

closely related lattice-Boltzmann (LB) [33–37] and lattice-gas automaton (LGA) [38,39] models that are defined on a regular lattice or grid have been extensively investigated. Although lattice-Boltzmann and lattice-gas cellular automaton models have been extended to a wide range of applications such as colloidal systems and multiphase flows in porous media, they have some disadvantages that are associated with the restriction of the dynamics to the streaming of 'particles' between adjacent nodes on a regular lattice. Another approach is to use particle-based simulation methods, similar to molecular dynamics, in which individual particles represent a volume of fluid that may vary in size, depending on the model, from a small cluster of atoms or molecules to macroscopic regions in a continuum solid or fluid. These off-lattice methods are manifestly Galilean invariant (unlike some lattice Boltzmann models). One of these methods, smoothed particle hydrodynamics (SPH), was originally invented to solve astrophysical problems [40–42], and it has been gradually modified for much smaller scale [22,43–46]. In SPH, the fluid is represented by overlapping weight functions, or smoothing functions, centered on the particles. The particles move with the local velocity of the fluid, and the acceleration of each particle is calculated from the local pressure gradient and the fluid density. The density at any point can be calculated from the positions of the particles that are within the range of the weight function, and the corresponding pressure is obtained from an equation of state. Other forces, such as those due to viscosity, which act in concert with the forces associated with the pressure gradient to determine the particle accelerations, can be estimated using the positions and velocities of neighboring particles,

the weight function and the derivatives of the weight function. The SPH method for mesoscopic applications is still under development, and quantitative relationships between model parameters and the macroscopic properties of the fluids that these models simulate are difficult to establish [47]. For more details on LB, LGA and SPH and their applications, please refer to some review articles and monographs. Typical review articles include the review on LB by Chen and Doolen [37], and Aidun and Clausen [48], the review on SPH by Monaghan [42,49], by Cleary et al. [43], and by Liu and Liu [50], and the review paper by Koumoutsakos on multiscale flow simulations using particles and particle methods [51]. Typical monographs include the book on LB by Sukop and Thorne [52], and by Guo and Shu [53], the book on LB and LGA by Wolf-Gladrow [54] and the book on SPH by Liu and Liu [22].

Dissipative particle dynamics (DPD) [55] is a relatively new mesoscale technique that can be used to simulate the behavior of fluids. As stated by Hoogerbrugge and Koelman [55], the DPD method is a "novel particle-based scheme combining the best of both MD and LGA simulations", which is "much faster than MD and much more flexible than LGA". In DPD simulations, the particles represent clusters of molecules that interact via conservative (non-dissipative), dissipative and fluctuating forces. Because the effective interactions between clusters of molecules are much softer than the interactions between individual molecules, much longer time steps can be taken relative to MD simulations. The longer time steps combined with the larger particle size makes it much more practical to simulate hydrodynamics using DPD than MD. DPD is particularly promising for the simulation of complex liquids, such as polymer suspensions, liquids with interfaces, colloids and gels. Because of the symmetry of the interactions between the particles in typical simulations, DPD rigorously conserves the total momentum of the system, and because the particle–particle interactions depend only on relative positions and velocities, the resulting model fluids are Galilean invariant. Mass is conserved because the same mass is associated with each of the particles, and the number of particles does not change. While DPD is not as computationally efficient as lattice Boltzmann simulations, it is a more flexible method that does not suffer from the numerical instability associated with many lattice Boltzmann applications. DPD facilitates the simulation of complex fluid systems on physically interesting and important length and time scales, including bio-particle and DNA filtering systems [56–58].

Español and Warren [59] and Marsh [60] established a sound theoretical basis for DPD based on statistical mechanics, and Groot and Warren obtained parameter ranges to achieve a satisfactory compromise between speed, stability, rate of temperature equilibration and compressibility [61]. Unlike traditional DPD methods that use a conservative pairwise force between particles that only depends on

their interparticle separation, the multi-body DPD (MDPD) model presented by Pagonabarraga and Frenkel [62] assumes that the conservative force also depends on the instantaneous local particle density, which in turn depends on the positions of many neighboring particles. Therefore, the conservative interaction is a many-body interaction.

It is possible to couple DPD with SPH for multiple scale simulations or develop a variety of 'hybrid' models that combine DPD and SPH concepts. Español [63] described a fluid particle dynamics (FPM) model that is a synthesis of DPD and SPH, and Español and Revenga [64] combined features from DPD and SPH to develop the smoothed dissipative particle dynamics (SDPD) model in which the Navier–Stokes (N–S) equation governing the system is discretized using SPH approximations while thermal fluctuations are included in a consistent way. Therefore, SDPD is a modified SPH model that has little in common with the original DPD method, except for the random forces representing the thermal fluctuations, which are an essential component of DPD simulations.

In this work, the dissipative particle dynamics shall be reviewed both in methodology and applications. The DPD methodology will be first introduced in Sect. 2, including the coarse-graining concept, governing equations, the time integration algorithms, the calculation of stress tensor, the determination of coefficients, and the common computational procedure in DPD simulation. In Sect. 3, some numerical aspects of DPD are discussed including the assessment of dynamic properties (such as viscosity and Schmidt number), solid boundary treatment for complex flow geometries, modification of conservative interaction potentials for modeling systems with co-existing liquid–gas–solid phases, and spring-bead chain model for simulating complex fluids (macromolecules such as DNA). In Sect. 4, recent applications of the DPD shall be reviewed with focus on micro drop dynamics, multiphase flows in pore-scale fracture network and porous media, movement and suspension of macromolecules in micro-channels, and movement and deformation of single cells. Section 5 gives some concluding remarks and future prospects.

## 2 Dissipative Particle Dynamics Methodology

### 2.1 Coarse-Graining

The classic molecular dynamics is a very important approach for investigating complex fluids such as polymers and macromolecules, and it is in principle capable of providing reliable results on all scales. As each particle in MD represents a true atom or molecule, MD can describe the dynamic behavior of a complex system with comprehensive details on every atom. MD simulations thus are usually limited to extremely

small time scales (nanoseconds) and length scales (nanometers) even if the state-of-art high performance computing techniques are used. However, most practical applications involve larger spatial and time scales. For example, polymers and many other materials frequently show a hierarchy of length scales and associated time scales. This requires a very large number of particles (and a very big number of degree of freedom) if using molecular dynamics simulation. To reduce the number of degrees of freedom, coarse-grained molecular dynamic techniques have been developed [58,65–67].

In the coarse-grained MD simulations, some trivial molecular details that do not affect the behavior at larger scales can be ignored, while the main features of concerned physics need to be effectively obtained. A general procedure in coarse-graining usually involves:

1. Defining the goal and determining the degree of coarse-graining;
2. Mapping atomistic model to coarse-grained model;
3. Interaction between the coarse-grained particles;
4. Reproducing target functions by the coarse-grained model;
5. Optimizing parameters/functions in the coarse-grained model, and
6. Conducting coarse-grained simulations.

The goal and degree of coarse-graining are usually application-driven and they describe the number of atoms/molecules in a typical particle in the coarse-grained model. This is closely related to the minimal features of the atomistic model that should be retained to reproduce the desired properties in the coarse-grained model. Mapping atomistic model to coarse-grained model is very important in defining the positions of coarse-grained particles and it directly influences the parameterization of the coarse-grained force field. The interaction between the coarse-grained particles is usually conducted with analytical functions (e.g., LJ potential in classic MD) or numerical functions of the positions of the coarse-grained particles.

Dissipative particle dynamics is such a coarse-grained molecular dynamics model, in which the particles represent clusters of molecules that interact via conservative (nondissipative), dissipative and fluctuating forces. As a coarse-grained MD model, DPD follows the above-mentioned coarse-graining procedure.

### 2.2 Governing Equations

In DPD models, a fluid system is simulated using a set of interacting particles. Each particle represents a cluster of small molecules instead of a single molecule. It is convenient to assume that all of the particles have equal masses, and use the mass of the particles as the unit of mass. Newton's second law governs the motion of each particle. The equation of motion for particle $i$ can therefore be expressed as

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \mathbf{f}_i = \mathbf{f}_i^{int} + \mathbf{f}_i^{ext}, \tag{1}$$

where $\mathbf{r}_i$ and $\mathbf{v}_i$ are the position and velocity vectors, and $\mathbf{f}_i^{ext}$ is the external force including the effects of gravity. In Eq. (1), the inter-particle force acting on particle $i$, $\mathbf{f}_i^{int}$, is usually assumed to be pairwise additive and consist of three parts: a conservative (non dissipative) force, $\mathbf{F}_{ij}^C$, a dissipative force, $\mathbf{F}_{ij}^D$, and a random force, $\mathbf{F}_{ij}^R$,

$$\mathbf{f}_i^{int} = \sum_{j \neq i} \mathbf{F}_{ij} = \sum_{j \neq i} \mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R. \tag{2}$$

Here, $\mathbf{F}_{ij}$ is the force on particle $i$ due to interaction with particle $j$, which is equal to $\mathbf{F}_{ji}$ in magnitude and opposite in direction. The symmetry of the interactions $\mathbf{F}_{ij} = -\mathbf{F}_{ji}$ ensures that momentum is rigorously conserved. The pairwise particle–particle interactions have a finite cutoff distance, $r_c$, which is usually taken as the unit of length in DPD models.

The conservative force, $\mathbf{F}_{ij}^C$, is a soft interaction acting along the line of particle centers, which is often given the form

$$\mathbf{F}_{ij}^C = a_{ij} w^C(r) \hat{\mathbf{r}}_{ij},$$
$$w^C(r) = \begin{cases} (1 - r) & r < 1.0, \\ 0 & r \geq 1.0, \end{cases} \tag{3}$$

where $a_{ij}$ is the maximum repulsion between particles $i$ and $j$, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, $r = r_{ij} = |\mathbf{r}_{ij}|$ and $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/r_{ij}$. Here, $w^C(r_{ij})$ is the weight function for the conservative force. It is noted that weight function corresponds to a soft potential, which allows much larger length and time scales in DPD. The weight function describes a repulsive force, which can well model the behavior of gas in confined spaces, but cannot be used to simulate liquid–gas–solid co-existing systems including the behavior of bubbly liquids, droplet dynamics and other important multiphase fluid flow processes.

The dissipative force, $\mathbf{F}_{ij}^D$, represents the effects of viscosity, and it depends on both the relative positions and velocities of the particles. The form usually used for this interaction in DPD simulations is

$$\mathbf{F}_{ij}^D = -\gamma w^D(r_{ij}) (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij}) \hat{\mathbf{r}}_{ij}, \tag{4}$$

where $\gamma$ is a coefficient, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ and $w^D(r_{ij})$ is a distance-dependent weight function. The random force, $\mathbf{F}_{ij}^R$, representing the effects of thermal fluctuations also depends on the relative positions of the particles, and it is defined as

$$\mathbf{F}_{ij}^R = \sigma w^R(r_{ij}) \xi_{ij} \hat{\mathbf{r}}_{ij}, \tag{5}$$

where $\sigma$ is a coefficient, $w^R\left(r_{ij}\right)$ is a distance-dependent weight function, and $\xi_{ij}$ is a random variable with a Gaussian distribution and unit variance. The dissipative force and random force also act along the line of particle centers and therefore also conserve linear and angular momentum.

As pointed by Español and Warren [59], in order to recover the proper thermodynamic equilibrium for a DPD fluid at a prescribed temperatures $T$, the coefficients and the weight functions for the random force and the dissipative force are related by

$$w^D(r) = \left[w^R(r)\right]^2, \qquad (6)$$

and

$$\gamma = \frac{\sigma^2}{2k_B T}, \qquad (7)$$

as required by the fluctuation-dissipation theorem. In Eq. (7), $k_B$ is the Boltzmann constant. All of the interaction energies are expressed in units of $k_B T$, which is usually assigned a value of unity. One simple, straightforward and commonly used choice is

$$w^D(r) = \left[w^R(r)\right]^2 = \begin{cases} (1 - r/r_c)^s & r < r_c \\ 0 & r \geq r_c \end{cases}, \qquad (8)$$

where $r_c$ is the cut-off distance of the dissipative and random force. In conventional DPD formulation, it usually takes the same value as the cut-off distance of the conservative force (usually unit value), but can vary to modify the dynamic properties in DPD simulation as will be shown later. $s$ denotes the exponent of the weighting function. It is reported by Fan et al. [68] that different $s$ can lead to different dynamic behavior of a DPD system. For conventional DPD formulation, $s = 2$. $w^D(r)$ and its gradient are both continuous at $r/r_c = 1$. In contrast, if $s < 1$, though $w^D(r)$ is still continuous, its gradient is not continuous at $r/r_c = 1$.

The random fluctuation force, $\mathbf{F}_{ij}^R$, acts to heat up the system, whereas the dissipative force, $\mathbf{F}_{ij}^D$, acts to reduce the relative velocity of the particles, thus removing kinetic energy and cooling down the system. Consequently, the fluctuating and dissipative forces act together to maintain an essentially constant temperature with small fluctuations about the nominal temperature $T$. Therefore, dissipative particle dynamics simulations are essentially thermostatted molecular dynamics simulations with soft particle–particle interactions.

### 2.3 Time Integration

The time integration algorithm is very important in DPD. Poor integration algorithms lead to serious problems such as equilibrium properties that depend on the magnitude of the time step. Early implementations of Eq. (1) in DPD made use of the Euler scheme

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t\mathbf{v}_i(t) \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \Delta t\mathbf{f}_i(t) \\ \mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i(\mathbf{r}_i(t + \Delta t), \mathbf{v}_i(t + \Delta t)), \end{aligned} \qquad (9)$$

where $\Delta t$ is the time step. The Euler scheme is not time reversible and it can lead to an energy drift in the system and hence it has been avoided in recent DPD research. Groot and Warren [61] used a modified version of the velocity-Verlet algorithm

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t\mathbf{v}_i(t) + \frac{1}{2}(\Delta t)^2\mathbf{f}_i(t) \\ \tilde{\mathbf{v}}_i(t + \Delta t) &= \mathbf{v}_i(t) + \lambda\Delta t\mathbf{f}_i(t) \\ \mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i(\mathbf{r}_i(t + \Delta t), \tilde{\mathbf{v}}_i(t + \Delta t)) \\ \mathbf{v}_i(t + \Delta t) &= v_i(t) + \frac{1}{2}\Delta t(\mathbf{f}_i(t) + \mathbf{f}_i(t + \Delta t)), \end{aligned} \qquad (10)$$

where $\tilde{\mathbf{v}}_i(t + \Delta t)$ is the prediction of the velocity at time $t + \Delta t$ and $\lambda$ is an empirically introduced parameter, which accounts for the effects of stochastic interactions. In this time integration algorithm, the velocity is first predicted to obtain the force and then corrected in the last step while the force is calculated only once during each integration step. It is found that for a velocity independent total force, the standard velocity-Verlet algorithm can be recovered at $\lambda = 1/2$ Groot and Warren reported that when simulating an equilibrium system with $\rho = 3.0$ and $\sigma = 3.0$, the optimum value of $\lambda$ is 0.65, which can lead to a considerable large time step to $\Delta t = 0.06$ without losing temperature balance [61].

Pagonabarraga et al. [69] proposed a leap-frog scheme which is self-consistent and can recover the correct equilibrium properties but needs iteration at each time step.

### 2.4 Stress Tensor

After obtaining the positions, velocities and forces on all DPD particles, the stress tensor, $\mathbf{S}$, is then calculated using the Irving–Kirkwood model [70] expressed by the equation

$$\mathbf{S} = -\frac{1}{V}\left[\sum_i \mathbf{u}_i\mathbf{u}_i + \frac{1}{2}\sum_{i \neq j}\mathbf{r}_{ij}\mathbf{F}_{ij}\right], \qquad (11)$$

where $V$ is volume and it is the reciprocal of the number density ($n$) of particles, $\mathbf{u}_i = \mathbf{v}_i - \bar{\mathbf{v}}(\mathbf{r})$ is the peculiar velocity of particle $i$, $\bar{\mathbf{v}}(\mathbf{r})$ is the stream velocity at position $\mathbf{x}$. The first term in the brackets is the kinetic (ideal gas) contribution describing momentum transfer and the second term is the contribution from the particle–particle interactions (or interparticle force). Just as expressed in Eq. (2), for simple DPD particles, the inter-particle force is the summation of conservative, dissipative and random force. For particles acting as a bead of molecular chains, the inter-particle force should include the total spring force on the particle.

The pressure, $p$, is obtained from the trace of the stress tensor,

$$p = -\frac{1}{3}tr\mathbf{S}. \tag{12}$$

## 2.5 Determination of Coefficients

The selection of coefficients in the DPD formulation directly influences the properties of the modeled DPD fluid (simulated properties). In order to match the simulated properties to the real properties and to maintain computational accuracy, parameters in DPD simulation need to be carefully chosen. Some coefficients can be determined by fitting the relevant data of the real fluid, some are selected to maintain the numerical accuracy in simulating simple cases with analytical solutions (e.g., Poiseuille flow). For complex system, just as pointed out by Fan et al. [68], there is no solid physical basis to determine the coefficients characterizing interaction strengths between different components.

### 2.5.1 Coefficients of Dissipative and Random Force

The coefficients of dissipative and random force ($\gamma$ and $\sigma$) are co-related by fluctuation-dissipation theorem, as expressed in Eq. (7). Therefore there is only one independent coefficient, and also the coefficient is closely related to noise amplitude of system temperature. Groot and Warren [61] ever tested the uniformly distributed random numbers and Gaussian distributed random numbers of the same variance and they found that there is no statistical difference between these two approaches. For temperature noise generated with uniformly distributed random numbers, increasing $\sigma$ beyond 8 can lead to rapidly growing temperature and unstable simulation. Taking $\sigma = 3$ with suitable parameters in the time integration algorithm (e.g., for the modified version of the velocity-Verlet algorithm expressed in Eq. (10), $\lambda = 0.5$ and $\Delta t = 0.04$) is usually a recommended value to get a reasonable balance between fast temperature equilibration, a fast simulation and a stable, physically meaningful system.

### 2.5.2 Time Step

It is found by Groot and Warren that, for the modified version of the velocity-Verlet algorithm expressed in Eq. (10), stable temperature control is obtained only when the term $\frac{1}{2}(\Delta t)^2 f_i(t)$ is included in the position update [61]. If this term is omitted, the simulation results are nearly as bad as the Euler algorithm. Empirically adjusting $\lambda$ for a given system (with specific $\rho$ and $\sigma$) can lead to a big time step without significant loss of temperature control. Groot and Warren reported that for a system with $\rho = 3$ and $\sigma = 3$ and an optimum value of $\lambda = 0.65$, the time step can be increased to $\Delta t = 0.06$ [61].

### 2.5.3 Repulsion Parameter

The repulsion parameter ($a$) for the conservative force (see Eq. 3) can be determined through matching the compressibility of the model fluid with real fluid. Groot and Warren found that for sufficiently high density ($\rho > 2$), a good approximation for pressure can be expressed as [61]

$$p = \rho k_B T + \alpha a \rho^2, \tag{13}$$

where $\alpha = 0.101 \pm 0.001$.

As the compressibility for a fluid can be expressed as

$$\kappa^{-1} = \frac{1}{k_B T}\left(\frac{\partial p}{\partial \rho}\right)_T, \tag{14}$$

it can be further written as

$$\kappa^{-1} = 1 + \frac{2\alpha a \rho}{k_B T}, \tag{15}$$

As the known compressibility of water under room temperature is approximately 16, it is found that $a = 75 k_B T/\rho$. Therefore for a given DPD system with specific temperature and density, the repulsion factor can be determined. For example, if $k_B T = 1$ and $\rho = 3$, the repulsion parameter (for DPD fluid mimicking the behavior of water) $a = 25$.

It should be noted that repulsion parameter $a$ for particles from different fluids can be different. For example, for particle interactions from the same kind of fluid $A$ or $B$, the repulsion parameter $a_{AA}$ may or may not equal $a_{BB}$. Again for particle interactions from two different fluids $A$ or $B$, $a_{AB}$ (or $a_{BA}$, where $a_{BA} = a_{AB}$) may also be different from $a_{AA}$ and $a_{BB}$ and in many cases, $a_{AB}$ can be taken as $\sqrt{a_{AA}a_{BB}}$. The different repulsion parameter can lead to different behavior of two fluids as in mixture or phase separation [61,71].

Also in DPD simulation, the interaction of fluid particles with particles from solid obstacles (solid particles) are necessary. However, there is no physical base on how the solid particles interact with each other, and interact with fluid particles. By taking a repulsion factor between solid particles ($a_{ww}$ or $a_w$, where $w$ means wall) different from that between fluid particles ($a_{ff}$ or $a_f$, where $f$ means fluid), it is feasible to get different repulsion factor between fluid and solid particles, $a_{wf}$. The interaction behavior thus can be quite different. For example, when modeling two-phase flow in micro channel or fractures, it is found that gradually increasing the ratio of $a_w$ to $a_f$ from 0 can lead to different wetting behavior from strong non-wetting to moderate non-wetting, weak wetting, moderate wetting, strong wetting effects, and even film flows [72].

## 2.6 Computational Procedure

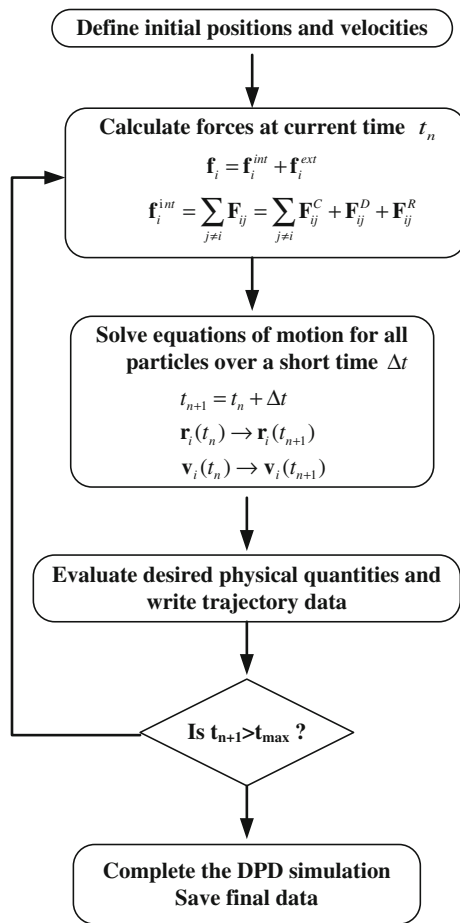DPD method is a coarse-grained molecular dynamics method, and its computational implementation is also similar to that

**Fig. 2** Computational procedure of a DPD simulation

in the classic MD. Figure 2 shows a typical computational procedure of a DPD simulation. As shown in Fig. 2, there are basically sequential stages: initialization, force computation, time integration and data analysis.

1. *Initialization* For the first run of a DPD simulation, it is necessary to initialize the coordinates of the DPD particles, their velocities and the target temperature for the simulation. Typically the DPD particles can be initially placed in a regular lattice spaced to give the desired density. They can also be injected into the computational domain according to a specific number density. The initial velocities are assigned with random directions and a fixed magnitude. It is preferred to initialize the velocity with appropriate Maxwell–Boltzman distribution for the specified temperature. However, a rapid equilibration renders the careful fabrication of a Maxwell–Boltzman distribution unnecessary. Initialization of DPD particle velocities is subjected to a number of conditions. For example, there is no overall momentum in any Cartesian direction, and the total kinetic energy is appropriate to the temperature specified.

2. *Force computation* In this stage, forces including the conservative force, dissipative force and random force are computed according to Eqs. (2)–(5). External forces such as the gravitational force can also be computed according to the specific physics.
3. *Time integration* After getting the forces, it is then possible to update the positions and velocities of all DPD particles according to a specific time integration algorithm.
4. *Data analysis* In this stage, desired physical quantities such as stress can be evaluated, and the trajectory data is then saved.

## 3 Numerical Aspects

### 3.1 Assessment of Dynamic Properties

Assume the radial pair distribution function, $g(r) \approx 1.0$, it is possible to derive the dynamic properties such as viscosity, diffusivity, and Schmidt number [61,68]. For a dissipative particle system with weight function expressed in Eq. (8) for the dissipative and random force, the dissipative viscosity can be expressed as a function of $s$ as follows

$$\eta^D = \frac{2\pi\gamma\rho^2 r_c^5}{15}\left(\frac{1}{s+1} - \frac{4}{s+2} + \frac{6}{s+3} - \frac{4}{s+4} + \frac{1}{s+5}\right). \tag{16}$$

It is noted that due to the soft interaction between DPD particles, the speed of momentum transfer is slow, and has the same order as the speed of particle diffusion. Therefore, the Schmidt number ($Sc$), defined as the rate of the speed of momentum transfer to the speed of particle diffusion, is about unity, which is much lower than $O(10^3)$ in a real fluid. For a typical DPD system, the dynamic viscosity is around $10^{-4}$ cP, which is also much lower than approximately 1 cP in real fluid. Therefore increasing the dynamic properties such as the Schmidt number and viscosity is usually necessary.

Figure 3 shows the influence of $s$ on the dissipative viscosity. It is clear that reducing $s$ can lead to considerably increasing viscosity. Table 1 shows the dynamic properties for a DPD system with $s = 1/2$, $s = 1$ and $s = 2.0$. It is found that different $s$ can lead to different dynamic properties. For example, for a given DPD system, the dynamic viscosity obtained with $s = 1/2$ is around 8 times the dynamic viscosity obtained with $s = 2.0$, and the Schmidt number is increased around 35.5 times when reducing $s$ from 2.0 to 1/2. Therefore reducing the exponential factor $s$ is an effective way to improve dynamic properties of the system with the same computational requirement.

Another approach to modify the dynamic properties of a DPD system is to change $r_c$ (cut-off distance for the dissipative, as expressed in Eq. (8)) and $\gamma$ (strength coefficient for the dissipative force as expressed in Eq. (4)), as the
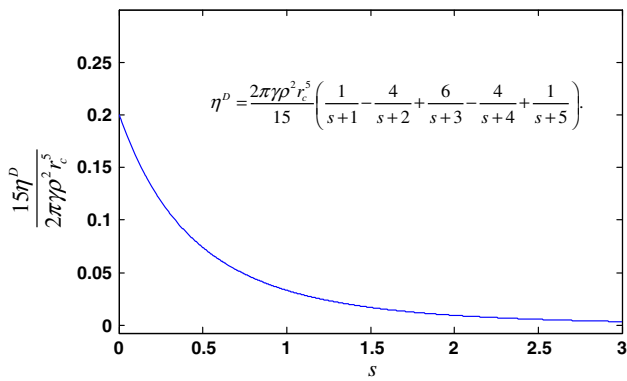
**Fig. 3** Viscosity as a function of $s$

$$\eta^D = \frac{2\pi\gamma\rho^2 r_c^5}{15}\left(\frac{1}{s+1} - \frac{4}{s+2} + \frac{6}{s+3} - \frac{4}{s+4} + \frac{1}{s+5}\right).$$

dynamic properties is dependent on $r_c$ and $\gamma$. Increasing $\gamma$ can result in larger fluctuation of thermal energy and requires good control of system temperature. Increasing $r_c$ is thus the most effective and easiest way to reduce the diffusivity and increase the dynamic viscosity and Schmidt number of the DPD system. However, increasing $r_c$ means enlarged computational cost. Therefore combining the modified weighting function and moderately increasing the cutoff radius for dissipative weighting function can enhance the dynamic viscosity and Schmidt number with reasonable computation costs. For example, for a DPD system with $\gamma = 4.5$, $\rho = 4.0$ and $k_B T = 1.0$, the influence of $r_c$ on the viscosity and Schmidt number for $s = 0.5$, 1 and 2 are shown on Figs. 4 and 5. It is clear that increasing $r_c$ can produce larger dynamic viscosity and Schmidt number. When $s = 0.5$ and $r_c = 1.88$, $Sc$ can reach about 1,000, which is of the same order as the Schmidt number of real fluid. In MD-like simulations, $r_c = 2.0 \sim 2.5$ is found to be satisfactory [68].

## 3.2 Solid Boundary Treatment

Just as in other CFD problems, solid boundary treatment is very important in DPD. To model the interaction between fluids and solid walls, both fluids and solid walls can be represented by DPD particles, which can be referred to as fluid particles and solid particles respectively. In DPD, a good solid boundary treatment algorithm should satisfy three requirements, (1) the fluid particles should not penetrate the solid walls unphysically, (2) there should not be large oscillation of physical variables in the boundary area, and (3) slip or no-slip boundary condition should be well implemented, either for fixed solid wall or moving solid obstacles [73,74].



**Fig. 4** Viscosity as a function of $r_c$ for different $s$



**Fig. 5** Schmidt number as a function of $r_c$ for different $s$

### 3.2.1 Reflection

During the simulation, some of the mobile particles that are used to represent the fluid(s) may penetrate into the wall particles because of the soft interaction between the DPD particles. In order to avoid such unrealistic penetration, one possible solution is to use a higher particle density for the walls or a larger repulsive force between the wall particles and fluid particles. This may cause large density oscillation in the boundary area.

**Table 1** Dynamic properties for DPD systems

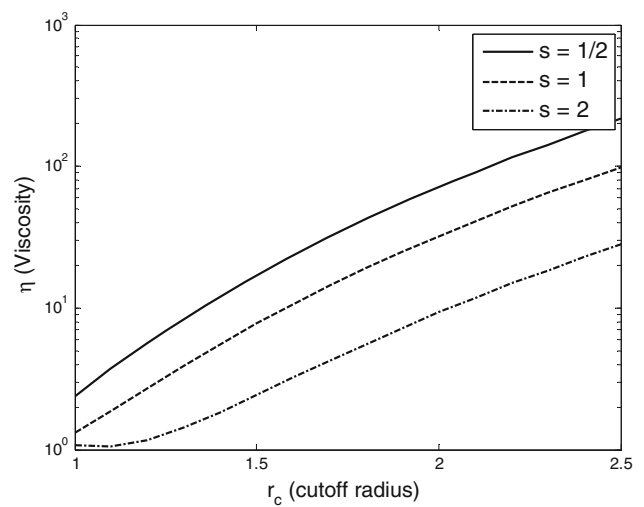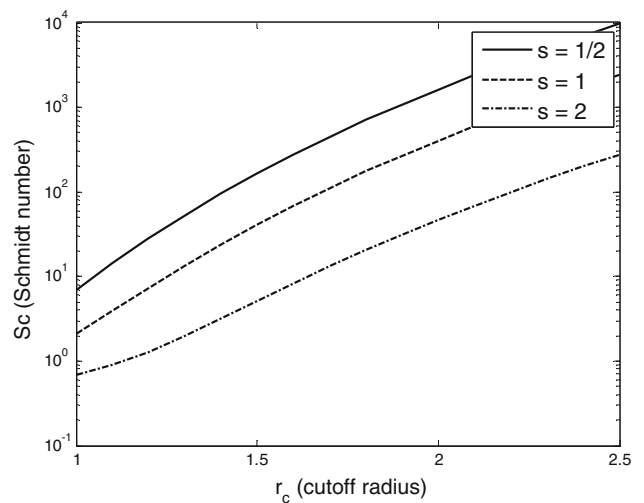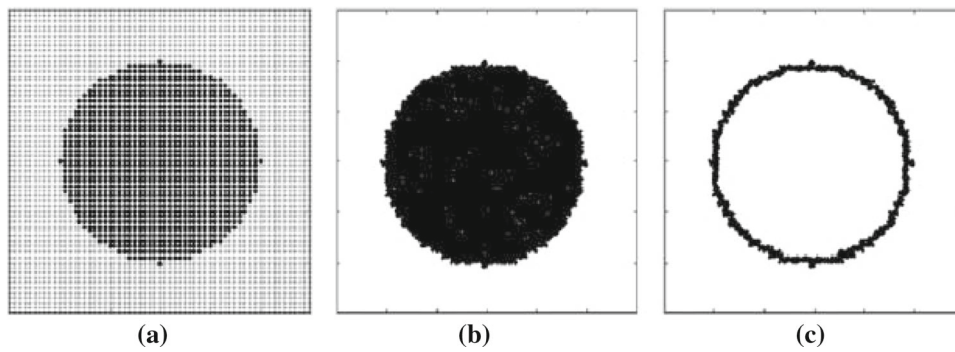| Properties | Conventional ($s = 2$) | Modified ($s = 1$) | Modified ($s = 1/2$) |
|---|---|---|---|
| Diffusivity, D | $\frac{45k_B T}{2\pi\gamma\rho r_c^3}$ | $\frac{9k_B T}{\pi\gamma\rho r_c^3}$ | $\frac{315k_B T}{64\pi\gamma\rho r_c^3}$ |
| Viscosity, $\eta$ | $\frac{\rho D}{2} + \frac{2\pi\gamma\rho^2 r_c^5}{1575}$ | $\frac{\rho D}{2} + \frac{\pi\gamma\rho^2 r_c^5}{225}$ | $\frac{\rho D}{2} + \frac{512\pi\gamma\rho^2 r_c^5}{51975}$ |
| Schmidt number, $Sc$ | $\frac{1}{2} + \frac{(2\pi\gamma\rho r_c^4)^2}{70875k_B T}$ | $\frac{1}{2} + \frac{(\pi\gamma\rho r_c^4)^2}{2025k_B T}$ | $\frac{1}{2} + \frac{(2\pi\gamma\rho r_c^4)^2}{1999k_B T}$ |

**(a)**        **(b)**        **(c)**

**Fig. 6** Illustration of the treatment of solid obstacles. **a** The cells in the entire computational domain are first labeled, "0" for fluid (void) cells and "1" for solid (obstacle) cells, **b** after equilibration, the DPD particles in the obstacle cells are frozen. **c** Only the frozen particles that are close to the fluid cells (within 1 DPD unit) are retained as boundary DPD particles (from [76])

Another frequently used approach in preventing unphysical penetration is based on reflection, in addition to the interactions between fluid and wall particles. Revenga et al. [73] investigated three different reflection models:

(a) specular reflection in which only the normal velocity component is reversed and the tangential velocity keeps unchanged (and therefore leading to free slip condition),
(b) bounce-back reflection in which all velocity components are reversed (same magnitude and opposite direction, and therefore leading to no-slip condition) and
(c) Maxwellian reflection in which particles are reflected back into the system according to Maxwell distribution.

It is noted that when implementing the Maxwell distribution, the velocities of particles that enter a thin layer next to the wall are selected randomly from the Maxwell distribution at temperature $T$ (thermal condition), with a zero mean corresponding to the zero fluid velocity at the boundary (no-slip condition). The velocity components can be reversed if the velocity points outward from the bulk fluid [68].

The treatment of solid boundaries by using frozen boundary particles and a thin reflecting boundary layer was found to be an effective way of implementing no-slip boundary conditions [68,75]. The thickness of the thin layer is selected to ensure that the probability of penetration is very low but the reflective layer occupies as little as possible of the fluid domain. In general, a thickness of 0.1 DPD unit is preferable for most applications. This thickness is small compared with the size of the fluid domain so it does not affect the bulk flow and it allows the fluid and wall particles to interact strongly enough to control the wetting behavior. On the other hand, it is large enough to prevent unphysical penetration. The implementation of no-slip boundary conditions with frozen wall particles and a thin boundary layer was found to be very flexible, especially for problems with complex geometries such as flow through porous media [76]).

### 3.2.2 Representation of Solid Grains

In DPD simulations, the effects of solid walls are usually simulated by using fixed particles to represent the solid matrix near to the solid–fluid interface. In the implementation, the entire computational domain can be discretized using a 'shadow' grid and grid cells are labeled "0" for regions occupied by pore spaces and "1" for solid filled regions (Fig. 6a). This simple identification of fluid and solid cells can be used to represent any arbitrary pore geometries including those determined from high-resolution X-ray and NMR tomography. The unit vectors normal to the solid–fluid interfaces, which define the local orientation of the interface, can be obtained by simply calculating the surface gradient from the indicator numbers (0 for liquid regions and 1 for solid regions). At the beginning of each DPD simulation, the particles are initialized and positioned randomly within the entire computational domain until a pre-defined particle number density is reached, and the system is then run to equilibrium using a DPD simulation with repulsive particle–particle interactions. The particles within the solid cells (marked as "1") are then 'frozen' (their positions are fixed) to represent the solid grains (Fig. 6b). The solid grains in porous media can occupy a considerable fraction of the entire computational domain, and hence the number of frozen particles representing the solids can be very large, particularly for low porosity media. Most of the frozen particles inside the solid grains are more than 1 DPD unit (or) away from the adjacent fluid cells. These particles do not contribute to the solid–fluid interactions and consequently they have no influence on the movement of the mobile DPD particles within the fluid cells. Therefore, only the frozen particles that are within 1 DPD unit (or) from the solid–fluid interface are retained as boundary DPD particles (Fig. 6c), and the rest of the particles further inside the solid grains are removed from the model domain. Figures 7 and 8 respectively show the representation of solid grains in a porous media and fracture network with frozen DPD particles within 1 DPD unit away

**Fig. 7** Representation of solid grains in a porous media with frozen DPD particles within 1 DPD unit away from the adjacent fluid cells
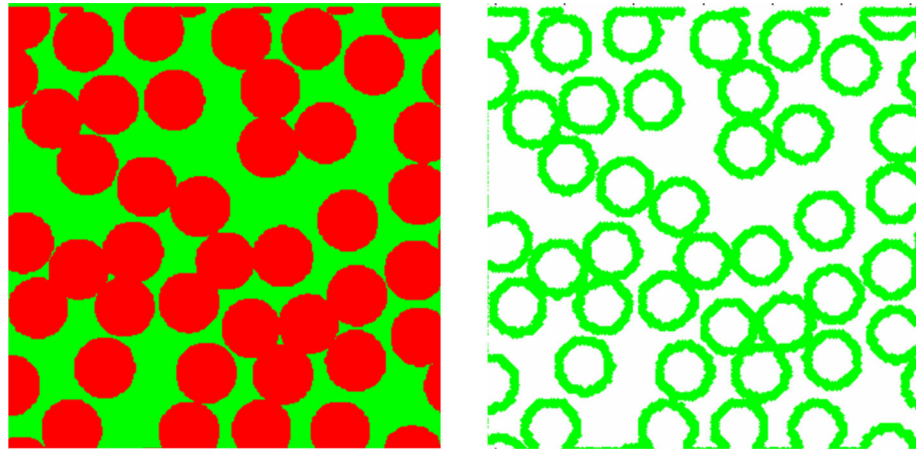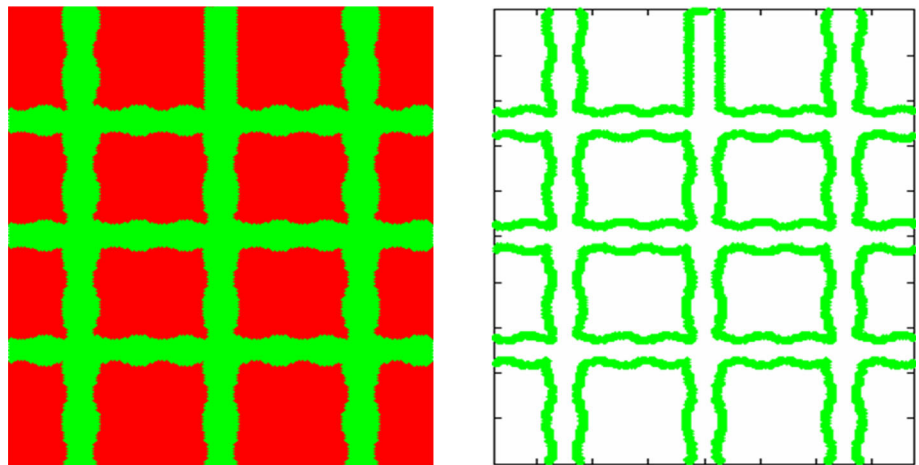


**Fig. 8** Representation of solid grains in a fracture network with frozen DPD particles within 1 DPD unit away from the adjacent fluid cells

from the adjacent fluid cells. It is clear that this treatment of solid grains is convenient to implement and suitable for arbitrary complex geometries.

### 3.2.3 Implementing Solid Boundary Condition

By using the above approach in representing solid grains and a suitable reflective model within a thin reflective boundary layer (see Fig. 9), it is possible to implement solid boundary conditions, either no-slip or slip. It is noted that this treatment of solid boundaries with frozen DPD particles within 1 DPD unit away from the adjacent fluid cells, and a thin reflective boundary layer in the fluid domain is effective in modeling complex solid obstacles, either fixed or movable [76].

### 3.3 Conservative Interaction Potential

#### 3.3.1 Constructing Conservative Interaction Potential

In conventional DPD implementations, a conservative force weighting function in a simple form $w^C(r) = 1 - r$ with a cutoff distance of $r_c$ (=1.0) has been used. Because the fluid generated by DPD simulations with this purely repulsive con-

servative force is a gas, it cannot be used to simulate the flow of liquids with free surfaces, the behavior of bubbly liquids, droplet dynamics and other important multiphase fluid flow processes. A direct solution of this problem is to include a long-range attractive component in $w^C(r)$. Like the repulsive component, the attractive component should also be a soft interaction to retain the advantages of the DPD method, and at short particle separations the repulsive component should be strong enough, relative to the attractive component to prevent the particle density from becoming too high. Moreover, the magnitude of the conservative force weight function and the location of the transition point from repulsion to attraction should be easily adjustable to allow the behavior of different fluids to be simulated.

Based on such considerations, Warren developed a many-body DPD (MDPD) for modeling vapor–liquid co-existing problems [77]. In MDPD, the conservative force can be expressed as

$$\mathbf{F}_{ij}^C = a_{ij}^A w^A(r)\hat{\mathbf{r}}_{ij} + b_{ij}^R w^R(r)\hat{\mathbf{r}}_{ij}, \qquad (17)$$

where the first term in Eq. (17) is the attractive force between particles $i$ and $j$, and the second term is the repulsive force between particles $i$ and $j$. $w^A(r)$ and $w^R(r)$ stand for con-

**Fig. 9** Illustration of implementing solid boundary condition
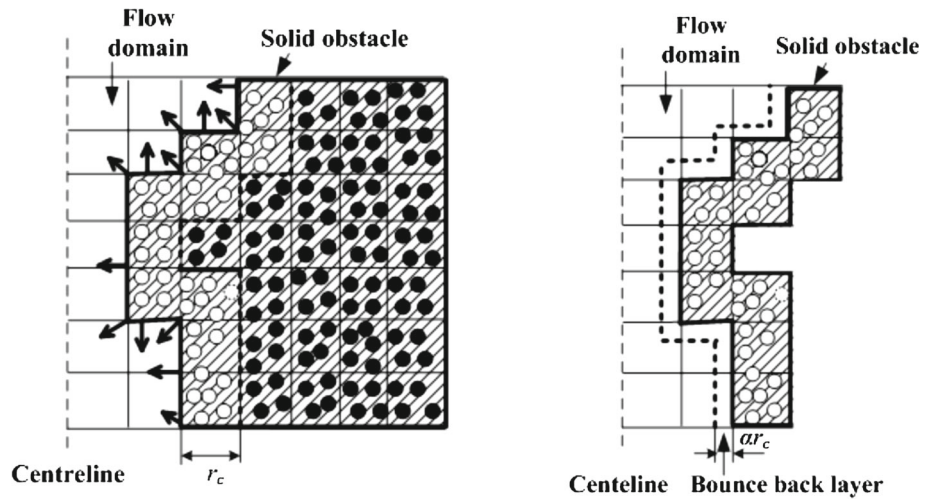


servative weight functions with different cut-off distance $r_A$ and $r_R$ for the attractive and repulsive force between interacting DPD particles. $a_{ij}^A$ and $a_{ij}^R$ are the corresponding strength coefficients for attraction and repulsion.

It is possible to construct polynomials that include both short-range repulsion and long-range attraction with a single cutoff distance [78]. Another approach is to combine commonly used SPH smoothing functions with different interaction strengths and cutoff distances to construct a particle–particle interaction potential. The most commonly used smoothing function in SPH [78] is the cubic spline,

$$W(r) = W(r, r_c) = \begin{cases} 1 - \frac{3}{2}\left(\frac{2r}{r_c}\right)^2 + \frac{3}{4}\left(\frac{2r}{r_c}\right)^3 & 0 \le \frac{2r}{r_c} < 1 \\ \frac{1}{4}\left(2 - \left(\frac{2r}{r_c}\right)\right)^3 & 1 \le \frac{2r}{r_c} < 2 \\ 0 & \frac{2r}{r_c} \ge 2, \end{cases}$$
(18)

where $r_c$ is the cutoff distance (corresponding to the smoothing length, $h$, in SPH) of the smoothing function. For the cubic spline function, $r_c = 2h$. In SPH, the function $W(r)$ in Eq. (18) is multiplied by a coefficient, $C$, so that the normalization requirement is satisfied [78]. The normalization coefficient, $C$, has values of $2/3h$, $10/7\pi h^2$ and $1/\pi h^3$ in the one-, two- and three-dimensional space. The cubic spline function defined in Eq. (18) is a non-negative, monotonically decreasing function, and it is smooth at both the origin and the cutoff.

One way of obtaining particle–particle interactions with the required short-range repulsive and long-range attractive form is to use a sum of spline functions multiplied by an interaction strength coefficient $a$,

$$U(r) = a\left(AW_1(r) - BW_2(r)\right) = a\left(AW_1(r, r_{c1}\right.$$
$$\left. - BW_2(r, r_{c2})\right),$$
(19)

to define the particle–particle interaction potentials, where $W_1(r)$ is a cubic spline with a cutoff length of $r_{c1}$, $A$ is the coefficient for $W_1(r)$, $W_2(r)$ is a cubic spline with a cutoff length of $r_{c2}$ and $B$ is the coefficient for $W_2(r)$. $W_1(r)$ and $W_2(r)$ are non-normalized shape functions given in Eq. (18). The DPD conservative particle–particle interaction forces are given by

$$\mathbf{F}_{ij}^C = \frac{-dU(r)}{dr}\hat{\mathbf{r}}_{ij}.$$
(20)

In DPD simulations, all particle–particle interaction potentials can have the same shapes with different interaction strengths for different particle–particle interactions. For example for fluid 1, the particle–particle potentials are defined by

$$U_{11}(r) = a_{11}\left(AW_1(r, r_{c1}) - BW_2(r, r_{c2})\right).$$
(21)

If a second fluid component is present, then the particle–particle interactions for that fluid are given by

$$U_{22}(r) = a_{22}\left(AW_1(r, r_{c1}) - BW_2(r, r_{c2})\right),$$
(22)

and the interactions between pairs of particles representing different fluids are given by

$$U_{12}(r) = U_{21}(r) = a_{12}\left(AW_1(r, r_{c1}) - BW_2(r, r_{c2})\right).$$
(23)

In Eqs. (21)–(23), $a_{ij}$ is the interaction strength between two particles representing component $i$ and component $j$ respectively.

A variety of functions can be obtained by using different combinations of $A, r_{c1}, B, r_{c2}$. For example, taking $A = 2.0, r_{c1} = 0.8$ and $B = 1.0, r_{c2} = 1.0$, and $a = 1$ (see Eq. 19), a function with positive and negative components is obtained. $AW_1(r)$, $-BW_2(r)$, and the resulting function
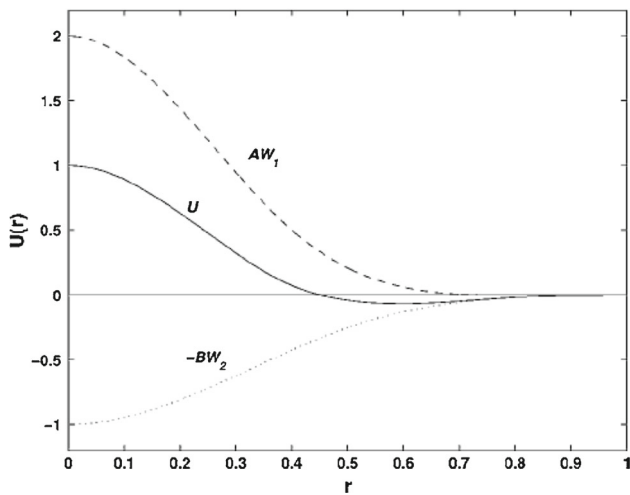
Fig. 10 Construction of a particle–particle interaction potential, $U(r)$, that is repulsive at short distances, attractive at intermediate distances and zero at large particle separation, from two cubic spline functions, $AW_1(r)$ and $BW_2(r)$ (from [71])
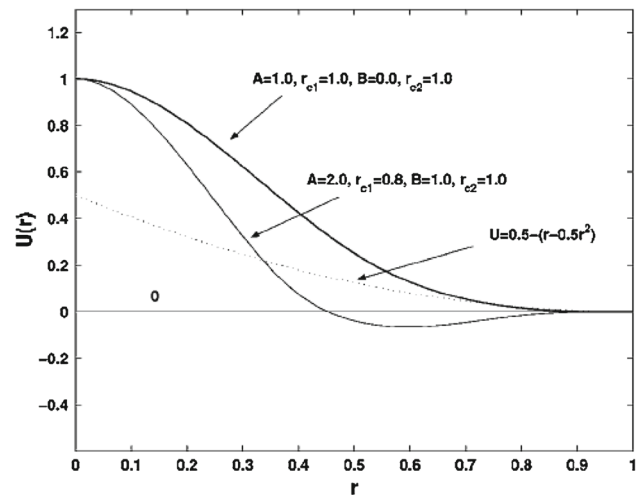


Fig. 11 Cubic spline potential functions, $U(r) = W_1(r, 1.0)$, $U(r) = 2W_1(r, 0.8) - W_2(r, 1.0)$ and the conventional DPD potential function, $U(r) = 0.5 - (r - 0.5r^2)$ (from [71])
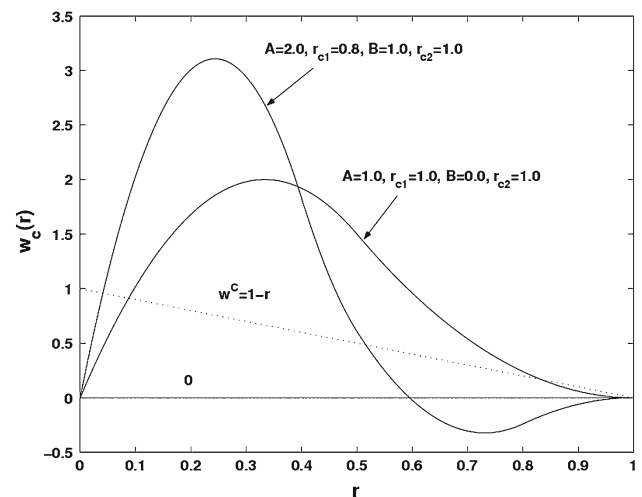


Fig. 12 Cubic spline conservative force weight functions and the conventional DPD conservative force weight function (from [71])

$U(r) = AW_1(r) - BW_2(r)$ is shown in Fig. 10. The figure shows that the function $U(r)$ is positive at the origin, gradually decreases, and then becomes negative at $r = 0.4529$. After reaching a minimum, $U(r)$ begins to increase until $U(r) = 0$ at $r = 1.0$. $U(r)$ is smooth at the origin and at the point $r = 1.0$. If $A = 1.0$, $r_{c1} = 1.0$ and $B = 0.0$, the resulting function $U(r)$ is the cubic spline expressed in Eq. (18), which is non-negative everywhere (see Fig. 11).

The spline function $W(r)$ describes a purely repulsive interaction and its negative counterpart $-W(r)$ describes a purely attractive interaction. The parameters $A$ and $B$ can be regarded as the strengths of the repulsive and attractive interactions. Different interaction strengths with corresponding cutoff distances generate different potential functions, $U(r)$, and corresponding weight functions, $w^C (= -U'(r))$, which can be used to simulate different phenomena. The two SPH cubic spline potential functions $U(r) = 2W_1(r, 0.8) - W_2(r, 1.0)$ (obtained by using $A = 2.0$, $r_{c1} = 0.8$, $B = 1.0$ and $r_{c2} = 1.0$, and $a = 1.0$) and $U(r) = W_1(r, 1.0)$ (obtained by using $A = 1.0$, $r_{c1} = 1.0$, $B = 0.0$ and $r_{c2} = 1.0$) as well as the conventional potential function $0.5 - (r - 0.5r^2)$ (corresponding to the conventional weight function $1 - r$) are shown in Fig. 11. The corresponding conservative force weight functions (or shape functions) are shown in Fig. 12.

Figure 12 shows that the conventional DPD conservative force weight function is non-negative and describes a purely repulsive interaction. Similarly the weight function obtained using $A = 1.0$, $r_{c1} = 1.0$ and $B = 0.0$ is also a purely repulsive non-negative function. While the weight function resulting from using $A = 2.0$, $r_{c1} = 0.8$, $B = 1.0$ and $r_{c2} = 1.0$ is a function with positive and negative sections, which corresponds to an interaction with short-range repulsive and

long-range attractive characteristics. The conventional DPD weight function is a monotonically decreasing function of the inter-particle separation with a constant negative (repulsive) slope whereas the new weight functions have regions with both positive (attractive) and negative slopes. Moreover, the derivatives of the new weight functions are smoother than the standard DPD weight functions.

Comparing Eqs. (17) and (19), it is found that Liu's approach is actually is equivalent to MDPD. In both approaches, the conservative force is divided into two components of attraction and repulsion. The attractive force and repulsive force between two interacting particles are associated with different cut-off distances and different strength coefficients for modeling the properties of different fluids. Different from MDPD, in which the basic form of conserv-

ative weight function is of the conventional form ($w^C(r) = 1 - r$), Liu's modified DPD approaches can use different polynomials (including the linear polynomial $1 - r$) to construct the interaction potential (or weight function).

### 3.3.2 Pressure–Density Relation

The combination of the attractive and repulsive interactions in the cubic spline potential makes it possible to simulate systems with co-existing liquid and gas phases and liquid–gas phase transitions. For a DPD system with attractive and repulsive interactions, the pressure–density relation can be numerically calculated. The fluid pressure can be calculated as a function of density from the particle–particle interactions using the virial theorem to obtain a numerical equation of state [25, 26]. Because the random and dissipative forces have average values of zero, they do not contribute to the virial pressure [61], and the total pressure is given by

$$P = P_k + \frac{\rho}{3} \sum_{j<i} (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{F}^C_{ij}, \tag{24}$$

where $P_k$ is the kinetic contribution ($P_k = \rho k_B T$, where $\rho$ is the fluid density).

The van der Waals (vdW) equation of state can also be used to model co-existing liquid and gas phases and liquid–gas phase transitions. The formulation of the van der Waals equation was motivated by the idea that short range repulsive forces lead to an effective volume for the gas molecules, which reduces the average free volume per molecule from $v$ to $v - \bar{b}$ and long range attractive forces reduces the pressure from $k_B T/(v - \bar{b})$ to $k_B T/(v - \bar{b}) - \bar{a}/v^2$. The resulting equation, $(p + \bar{a}/v^2)(v - \bar{b}) = k_B T$, provides a quantitative model for the phase behavior of simple fluids. In particular, for van der Waals fluids (model fluids described by the van der Waals equation) gas and liquid phases may coexist in a non-zero region of the $(p, v, T)$ or $(p, \rho, T)$ parameter space (depending on the coefficients $\bar{a}$ and $\bar{b}$) where $\rho$ is the average fluid density. The van der Waals fluid is the classic example of a fluid with co-existing liquid and gas phases and liquid–gas phase transitions. The equation of state for a van der Waals fluid can be expressed in the form [79]

$$p = \frac{\rho k_B T}{1 - \rho \bar{b}} - \bar{a} \rho^2, \tag{25}$$

where $\bar{a}$ controls the strength of the attractive force, and $\bar{b}$ is related to the size of the particle. This equation of state can be obtained from the macroscopic free energy density for interacting particles with short range repulsive interactions and long range attractive interactions in the mean field (infinite interaction range) limit [80]. Giving $\bar{a}$ and $\bar{b}$, it is easy to plot the pressure–density relation for a constant tem-
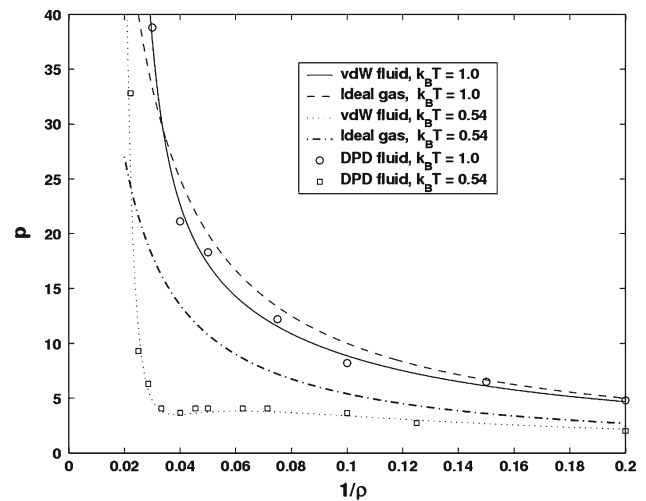


**Fig. 13** Pressure–density relations for a van der Waals fluid with $\bar{b} = 0.016$, $\bar{a} = 1.9b$, and a DPD fluid with $A = 2.0$, $r_{c1} = 0.8$, $B = 0.95$, $r_{c2} = 1.0$. The temperatures are $k_B T = 1$, and $k_B T = 0.54$ respectively (from [71])

perature. Figure 13 shows the pressure–density relations for a van der Waals fluid with $\bar{b} = 0.016$ and $\bar{a} = 1.9b$ while the temperatures are $k_B T = 1$ and $k_B T = 0.54$ respectively.

The pressure–density–temperature relationship for a wide range of fluids can be represented quite well by a van der Waals equation of state over a limited part of the parameter space. By tuning the parameters $\bar{a}$ and $\bar{b}$, it is possible to obtain a van der Waals equation of state for DPD fluids. Figure 13 shows the pressures calculated at a number of densities for a DPD fluid with $A = 2.0$, $r_{c1} = 0.8$, $B = 0.95$, $r_{c2} = 1.0$, and for $k_B T = 1$ and $k_B T = 0.54$. The simulations were implemented by placing different number of DPD particles into a box of size $10 \times 10 \times 10$ with periodic boundary conditions in all three directions to model the effects of different global densities, $\rho = n/1{,}000$. The averaged total pressure was calculated using the virial theorem relationship given in Eq. (24). The pressure calculated for the DPD fluid at different densities can be represented well by the van der Waals equation, as Fig. 13 shows.

### 3.4 Spring-Bead Chain Models

In the DPD model, a macromolecule chain can be represented by a chain of particles (beads) connected by springs. Similar to fluid particles (for modeling simple fluids) that can be thought of as a small regions of fluid, the macromolecule beads can be regarded as polymeric chain segments consisting of number of monomeric units. The macromolecule beads exchange momentum with each other according to the spring force and other ordinary DPD interactions. Hydrodynamic and thermodynamic interactions between the macromolecule and solvent then emerge naturally in these simu-

lations. Numerous simulations have verified that the DPD model can capture many essential physical phenomena of the macromolecule systems.

A number of spring-bead chain models have been used in polymer rheology as the coarse-grained models of macromolecules. Typical of them are the worm-like chain (WLC) model and finitely extendable nonlinear elastic (FENE) model. In the WLC model, the spring force law of a wormlike chain segment can be expressed as

$$\mathbf{F}_{ij}^{S} = -\frac{k_B T}{4\lambda_p^{eff}} \left[ \left(1 - \frac{r_{ij}}{l}\right)^{-2} + \frac{4r_{ij}}{l} - 1 \right] \hat{\mathbf{r}}_{ij}, \quad (26)$$

where $l$ is the maximum length of one chain segment and $\lambda_p^{eff}$ is the effective persistence length of the chain. If the total length of the chain is $L$ and the number of bead in the chain is $N_b$, $l = L/(N_b - 1)$. It was found that the mechanical properties of DNA molecules in an aqueous solution can realistically be modeled by the wormlike chains [81,82].

The spring force law of a FENE chain segment can be expressed as the following equation

$$\mathbf{F}_{ij}^{S} = -\frac{H\mathbf{r}_{ij}}{1 - (r_{ij}/r_m)^2}, \quad (27)$$

where $H$ is the spring constant. $r_m$ is the maximum length of one FENE chain segment. From Eq. (27), we can see that the spring force increases intensely and approaches infinity when $r_{ij}/r_m$ approaches 1. As a result, the distance between two neighboring beads in FENE chain should be less than $r_m$.

## 4 Applications

Like other Lagrangian particle based methods such as SPH, DPD models have special advantages over the traditional grid based methods in modeling fluid flow in domains with complex solid boundaries. They do not require explicit and complicated interface tracking algorithms, and thus there is no need to explicitly track the material interfaces, and processes such as fluid fragmentation and coalescence can be handled without difficulty. The effectiveness of DPD in modeling complex physics and reproducing the continuum hydrodynamic behavior has been demonstrated in various applications [59,75,83]. It is noted that a DPD model should conform to the Navier–Stokes equations on scales that are large enough for hydrodynamic (continuum) concepts to be valid (on scales large enough for the effects of both the mean free path of discrete particles and their thermal fluctuations to be negligible), providing that the time step in the integration scheme is small enough to ensure accurate integration [60]. A number of previous investigations have shown that the results obtained from DPD simulations are in good agreement with the flow behaviour predicted by the Navier–Stokes equations for a variety of fluid flows [71,75].

As a coarse-grained molecular dynamics method, DPD is attractive in modeling the hydrodynamic behavior of mesoscopic complex fluids. Therefore, since its invention, the DPD method has been extended to many applications including colloidal suspensions [84], surfactants [85], dilute polymer solutions [86], biological membranes [87], macromolecular dynamics [75,88] and many others [9]. Most of the earlier applications focus on the equilibration process of complex fluids including the aggregation of polymer and surfactant, the mixture or phase separation and morphology evolution of complex fluids with multi-components or multiphases. Recently DPD method is popular in modeling the dynamic flow process of mesoscopic complex fluids including liquid drop dynamics (drop formation, oscillation, coalescence, collision, impacting, and spreading) and the saturated or unsaturated flows in mesoscopic structures (microchannels, fractures and porous media).

Here we concentrate on the following areas

- multiphase drop dynamics
- multiphase flow in micro-channels and fractures
- movement and suspension of macromolecules
- movement and deformation of a single cell

### 4.1 Micro Drop Dynamics

Characterization of fluid flows in microfluidic devices has increasingly becoming a very important topic since the fluidic behavior in MEMS is very different from what observed in daily life. Flows in microfluidic devices usually involve small or ignorable inertial force, but dominant viscous, electrokinetic and surface effects especially when the surface-to-volume ratio increases [89]. Analytical or semi-analytical solutions for microfluidics are generally limited to a very few simple cases, whereas experimental studies are usually expensive. Numerical simulation of flows in microfluidic devices, as an effective alternate, has been attracting more and more researchers. However, simulation of microfluidic devices is not easy due to the involved complex features including movable boundaries (free surfaces and moving interfaces), large surface-to-volume ratio, and phenomena due to microscale physics. Numerical studies with reliable models are needed to develop a better understanding of the temporal and spatial dynamics of multiphase flows in microfluidic devices.

Drop formation and break-up in micro/nano scales are fundamentally important to diverse practical engineering applications such as ink-jet printing, DNA and protein micro-/nano-arraying, and fabrication of particles and capsules for controlled release of medicines. Numerical studies provide an effective tool to improve better understanding of the

inherent physical dynamics of drop formation and breakup. Computational models for drop formation and breakup in micro/nano-scales must be able to handle movable boundaries such as free surfaces and moving interfaces, large density ratios, and large viscosity ratios. These requirements together with microscale phenomena and possible complex boundaries (fluid–fluid–solid contact line dynamics and fluid–fluid interface dynamics) in microfluidic devices present severe challenges to conventional Eulerian-grid-based numerical methods such as finite difference methods and finite volume methods which require special algorithms to treat and track the interfaces. Algorithms based on Lagrangian-grid-based methods such as finite element methods have been shown to agree quantitatively with experimental measurements, but they are only capable of modeling the dynamics of formation of a single drop or the dynamics until the occurrence of the first singularity.

Some researchers have simulated drop dynamics for multiple component systems by using the DPD method with the conventional weight function for the conservative force [90]. However, the conventional conservative force is able to simulate liquid–liquid and liquid–solid interfaces for multiple component systems, but is not able to simulate liquid–gas interfaces for single component systems. Recently the DPD method is modified to model the solid–liquid–gas co-existing systems, either by using the many body DPD approach developed by Warren [77] or by using a new conservative interaction potential with long-distance attraction and short-range repulsion proposed by Liu et al. [71], or by other approaches which are able to describe the attraction and repulsion between interacting DPD particles. For example, Li et al. [91] investigated the 3D flow structures in a moving droplet on substrate by using the many body DPD, Zhang et al. [92] studied the movement of a droplet in a grooved channel by using Liu's conservative interaction potential. Merabia and Pagonabarraga developed a mesoscopic model for simulating the dynamics of a non-volatile liquid on a solid substrate and they analyzed the kinetics of spreading of a liquid drop wetting a solid substrate and the dewetting of a liquid fill on a hydrophobic substrate [93]. Based on mean-field theory, Tiwari and. Abraham proposed a DPD model for two-phase flows involving liquid and vapor phases [94]. The DPD model is validated by a number of numerical examples including the small- and large-amplitude oscillations of liquid drops.

Revisiting the new interaction potential in (19) with long-distance attraction and short-range repulsion, different parameter sets, $A, r_{c1}, B$ and $r_{c2}$, determine the shape of the interaction potential, and consequently the behaviour of the DPD fluid. It is possible to model micro drop dynamics (drop formation, oscillation, coalescence, collosion, impacting and spreading) with co-existing liquid–vapour phase use this interaction potential.
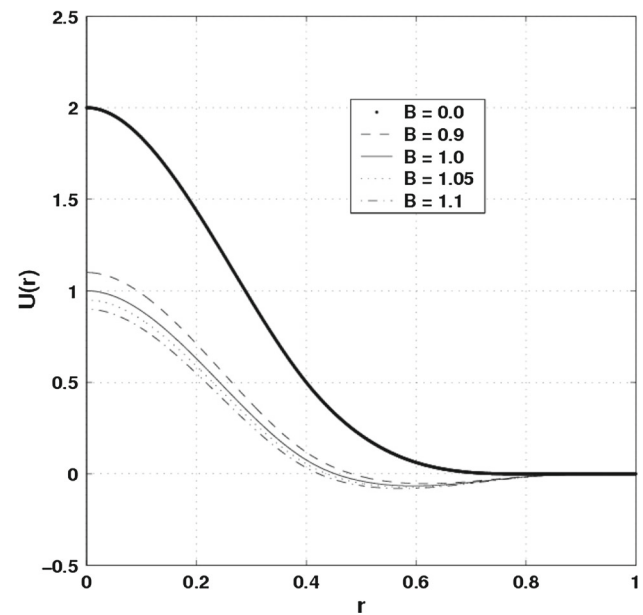


**Fig. 14** Cubic spline interaction potential functions, $U(r) = 2W_1(r, 0.8) - BW_2(r, 1.0)$ with different coefficients (from [71])

For example, in a DPD simulation of simple fluids, the coefficients associated with the fluctuating and dissipative forces can be $a = 18.75, \sigma = 3.0$, and $k_B T = 1.0 (\gamma = 4.5)$. If the parameters for the interaction potential are selected as $A = 2.0, r_{c1} = 0.8$ and $r_{c2} = 1.0$, with several values for $B$, it is possible to investigate different DPD fluid behaviours resulted from different attractive effects. The particle–particle interaction potentials were given by $U(r) = 18.75 (2W_1(r, 0.8) - BW_2(r, 1.0))$, which were shown in Fig. 14. The fluids can demonstrate different behavior with different $B$. when $B = 0$, the interaction potential represents a repulsive force, which makes the DPD particle to fill the entire computational domain, as shown in Fig. 15. Since there is no attractive component in the particle–particle interaction, the DPD particles did not separate into liquid and gas phases, and they did not form a liquid drop (or drops). A small value of $B$, corresponding to weak long-range attraction between the DPD particles is not sufficient to induce phase separation. When the critical value for $B$ is reached, at a particular temperature, large density fluctuations will occur, and an additional small increase in $B$ will lead to slow phase separation. Figure 16 shows the particle distribution after equilibration with $B = 0.9$. The fluid forms a spherical liquid drop surrounded by dense gas particles. The size of the spherical liquid drop remained approximately constant, while the location of the drop centre shifted a little. This is expected since collisions between 'gas' particles and the drop will induce the drop to undergo Brownian motion, and combined evaporation and condensation will also result in random motion of the centre of mass).
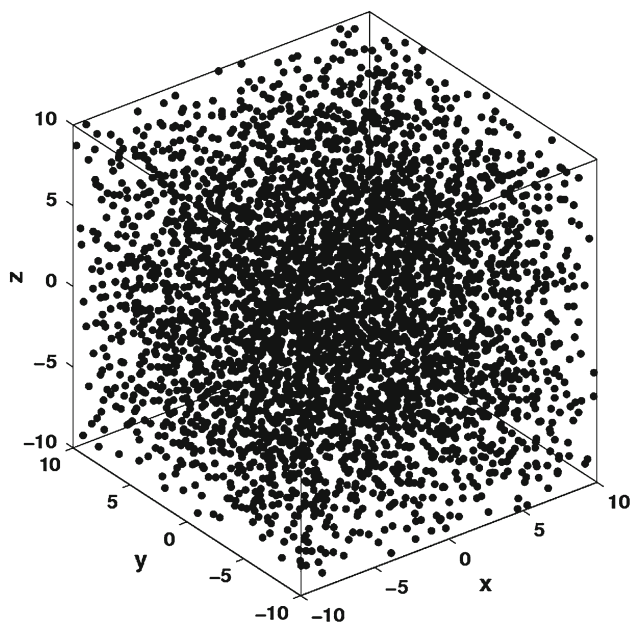
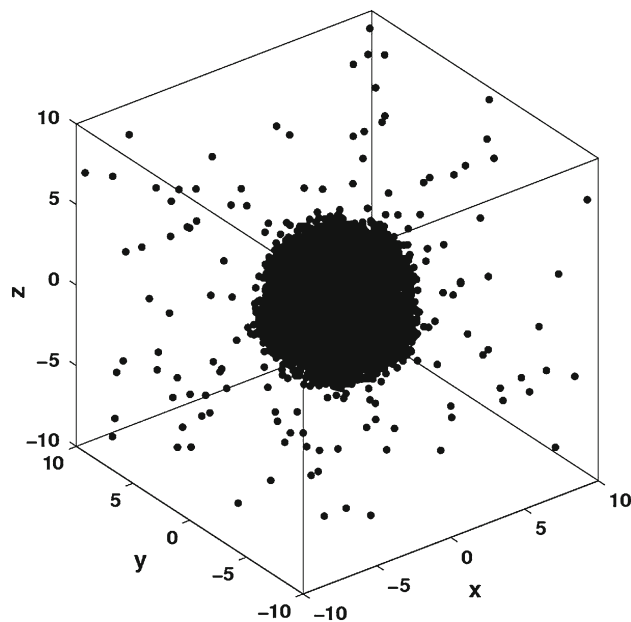**Fig. 15** Particle distribution obtained using the cubic spline potential $U(r) = 37.5W_1(r, 0.8)$ (from [71])



**Fig. 17** Particle distribution obtained using the cubic spline potential $U(r) = 18.75(2W_1(r, 0.8) - 1.0W_2(r, 1.0))$ (from [71])
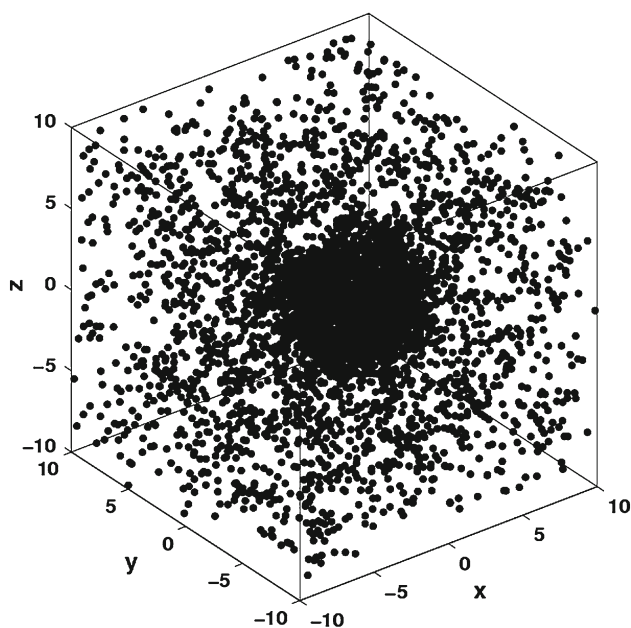


**Fig. 16** Particle distribution obtained using the cubic spline potential $U(r) = 18.75(2W_1(r, 0.8) - 0.9W_2(r, 1.0))$ (from [71])
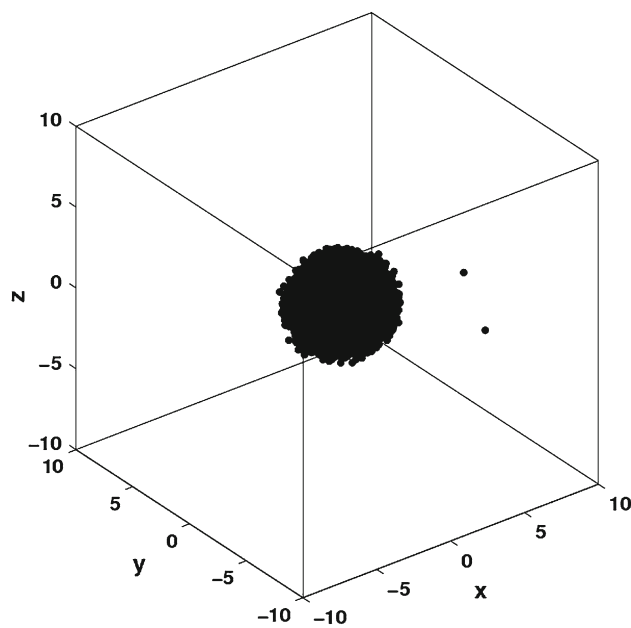


**Fig. 18** Particle distribution obtained using the cubic spline potential $U(r) = 18.75(2W_1(r, 0.8) - 1.05W_2(r, 1.0))$ (from [71])

Figure 17 shows the particle distribution after equilibration using $B = 1.0$. In this case, the fluid also forms a spherical liquid drop with sparse gas particles surrounding it. The liquid/gas density ratio is greater than 600. The shape of the liquid drop was stable, the drop size was almost constant, and the interface width is roughly equal to the interaction range. The number of surrounding gas particles was much smaller than that in the case illustrated in Fig. 16. Further increases in

$B$ result in stronger attractive effects in the interaction. Figure 18 shows the particle distribution after equilibration from a simulation with $B = 1.05$. The bulk fluid forms a stable spherical liquid drop. In this case the density of the gas is very small, the number of 'gas' particles fluctuates strongly and at some time there may be no gas particles at all in the relatively small volume used in this simulation. Additional increases in $B$ can result in different behaviour. Figure 19 shows the
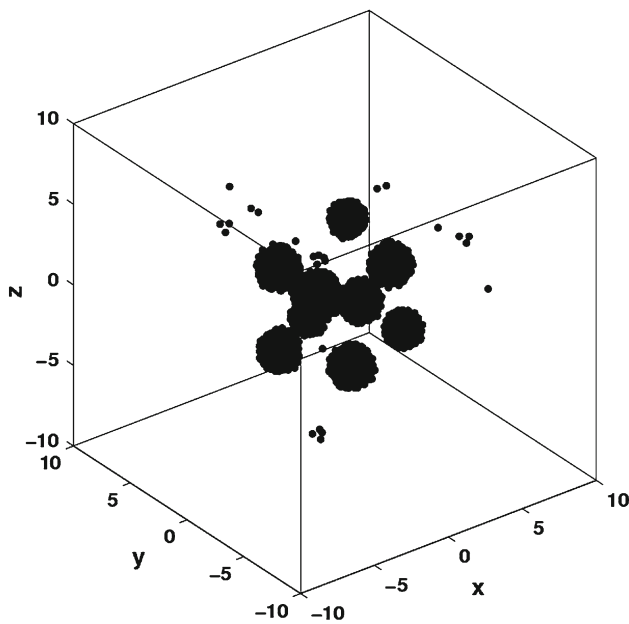
**Fig. 19** Particle distribution obtained using the cubic spline potential $U(r) = 18.75(2W_1(r, 0.8) - 1.1W_2(r, 1.0))$ (from [71])

particle distribution after equilibration using $B = 1.1$. The fluid underwent a phase transition but instead of forming a single liquid drop surrounding with gas particles, a number of small droplets formed. This can be expected due to the stronger particle–particle attraction, which resulted in a number of small droplets. Eventually a single drop should be formed by a process similar to Ostwald ripening [95] in which particles evaporate more rapidly from small droplets and condense more rapidly on large ones.
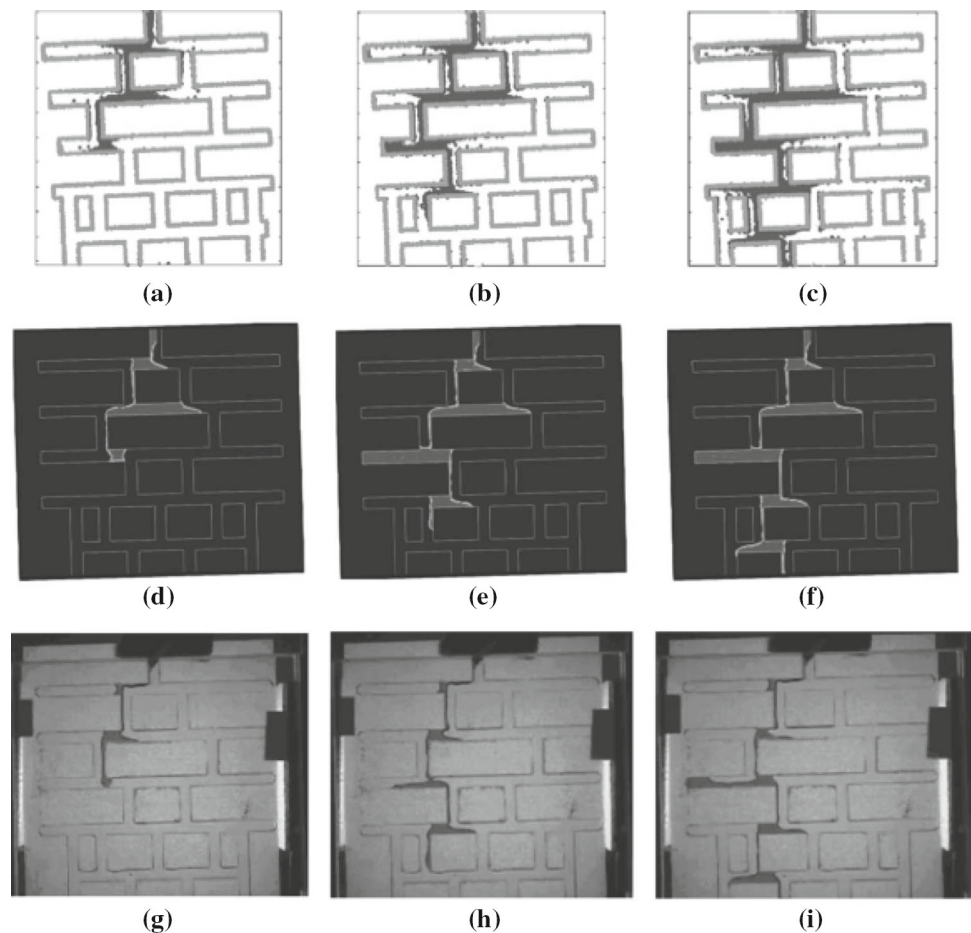
### 4.2 Multiphase Flows in Pore-Scale Fracture Network and Porous Media

Pore-scale, multiphase fluids in contact with solid surfaces are important in almost all areas of science and technology including nuclear reactor heat exchangers, lubricated pipeline transport, manufacturing of multilayer films and fibers, chemical reactors and separators, coating systems, enhanced oil and gas production, and environmental remediation [96,97]. They involve complex physics of fluid–fluid–solid contact line dynamics and wetting behaviors which are closely related to the inter particle and intra molecular hydrodynamic interactions of the concerned multiple phase system. For example, unsaturated fractures in the vadose zone are very important for groundwater recharge, fluid motion and contaminant transport, and flow through fractures can lead to exceptionally rapid movement of liquids and associated contaminants [98,99]. The physics of fluid flows in unsaturated fractures is still poorly understood due to the complexity of multiple phase flow dynamics.

Experimental studies of fluid flow in fractures are limited [100]. In computer simulations it is usually difficult to take into account the fracture surface properties and microscopic roughness. Predictive numerical models can be divided into two general classes: volume-averaged continuum models (such as those based on Richard's equation) [101] and discrete mechanistic models [102]. Knowledge of the physical properties of the fluids and the geometry of the fracture apertures is required in both classes. Volume-averaged continuum models are more suitable for large-scale systems, and they usually involve the representation of fractures as porous media with porosity and permeability parameters adjusted to mimic flow within fractures. However, volume-averaged continuum models are unable to describe the details of flow dynamics in fractures, they do not reproduce the spatio–temporal complexity of multiphase fluid flow in fractures, and they often fail to predict the rapid fluid motion and contaminant transport observed in the fractured vadose zone. Small-scale studies with discrete mechanistic models are needed to develop a better understanding of the temporal and spatial dynamics of fracture flows. However, the complexity of fracture flow dynamics makes it difficult to develop successful numerical models for fluid flows in fracture networks. A broadly applicable model must be able to simulate a variety of phenomena including film flow with free surfaces, stable rivulets, snapping rivulets, fluid fragmentation and coalescence (including coalescence/fragmentation cascades), droplet migration and the formation of isolated single-phase islands trapped due to aperture variability.

Realistic mechanistic models for multiphase fracture flows must be able to handle moving interfaces, large density ratios (e.g., $\approx 1,000{:}1$ for water and air), and large viscosity ratios (e.g., $\approx 100{:}1$ for water and air). These requirements combined with the complex geometries of natural fractures present severe challenges to mechanistic models. Grid-based numerical methods such as finite difference and finite volume methods and Eulerian finite element (FE) methods require special algorithms to treat and track the interface between different phases. These algorithms are usually complicated and fall into two general groups, interface tracking and interface capturing. Interface tracking algorithms generally use marker particles within grid cells intersected by the interface to identity the locations of interfaces [103,104]. The particles are then advected with the flow, and the positions of the interfaces can be determined from the particle positions. This approach is computationally expensive, especially for three-dimensional simulations, and often requires additional interface repairing techniques when the interface topology changes. Interface capturing algorithms are usually based on an 'indicator' field function with different values for different phases. The location of the interface can be determined from the indicator function, $f(\mathbf{x})$ where $\mathbf{x}$ is the position in

**Fig. 20** Sequential images of fluid motion into a micro channel network. The 3 figures in the *top row* (**a**–**c**) show the DPD simulation snapshots, the 3 figures in the *middle row* (**d**–**f**) show the VOF simulation results, and the 3 figures in the *bottom row* (**g**–**i**) show the experimental photographs at 3 equivalent stages (from [107])



(a)                                   (b)                                   (c)

(d)                                   (e)                                   (f)

(g)                                   (h)                                   (i)

the $D$-dimensional computational domain, which may have a specific value at the interface, or a range of values with a large gradient near the interface. The evolution of the moving interface can be obtained from the evolution of the indicator function. The volume of fluid (VOF) approach [105] is based on an indicator function that specifies how much fluid of each phase is contained in each of the grid cells. In the level-set (LS) function approach [106], the interface is a $D$−dimensional cut (contour) at $f = f^O$, through the $D+1$-dimensional surface $f(\mathbf{x})$. In most implementations, for two phase systems, $f(\mathbf{x})$, is positive in regions occupied by one phase, negative in regions occupied by the other, and $f^O = 0$. The VOF approach is robust and the mass loss/gain during a simulation is usually well controlled. But the captured interface usually spans several grid cells. In the LS approach, the interface is more sharply defined, but the loss/gain of mass during a simulation is larger.

There are a number of works in using the DPD method to model the multiphase flow in micro-channels or fractures with surface tension and wetting effects [72,92,107,108]. Figure 20 shows the DPD simulation of multiphase flow through a channel network together with the numerical study using a volume of fluid (VOF) method was presented by Huang et al. [109], and a flow experiment based on the same

channel network fabricated using polymethylmethacrylate. The flow patterns, penetration depths and formation of a quasi-steady state flow path during the late stages obtained from DPD simulation, VOF simulation and experiment are in general agreeable.

Figure 21 shows the DPD simulation of the infiltration process, together with numerical results from VOF simulation at four different stages [76]. For all stages, the injected liquid flows preferentially through the largest pore throats and through the vertical 'microfracture' where the capillary forces resisting flow are minimal. It is seen that although there are slight differences between the DPD and VOF simulations in several of the pore spaces, the overall flow patterns within the fracture are almost the same. The slight differences in these pore spaces also originate from the different ways used to represent obstacles in two approaches. The solid obstacles are represented using "solid" occupied grid cells in the VOF model, but they are represented using randomly distributed frozen particles in the DPD model. This leads to slight differences between the flow patterns simulated using the VOF and DPD models. In general, the visual comparison between these two simulations clearly reveals that these quite different approaches give essentially the same fluid dynamics in the fractured porous medium.

**Fig. 21** VOF (*top*) and DPD (*bottom*) simulation of non-wetting fluid infiltration into a fractured porous medium at four equivalent stages (from [76])
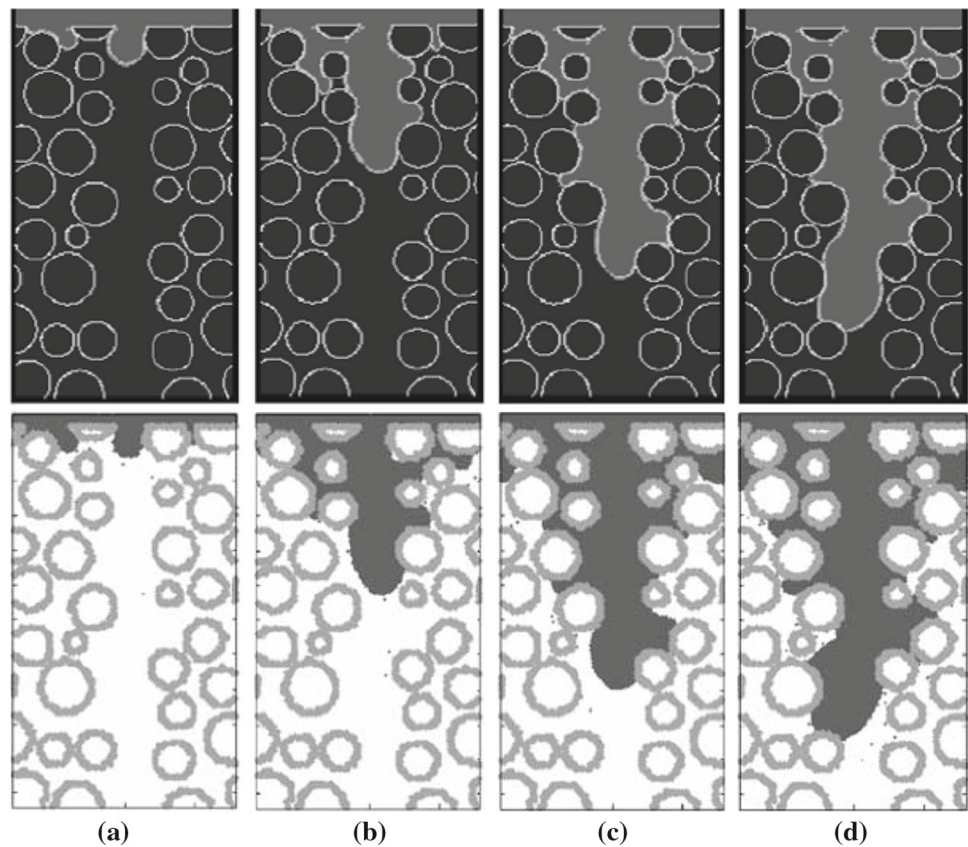


### 4.3 Movement and Suspension of Macromolecules in Micro-channels

Understanding the dynamic behavior of macromolecules, such as DNA, is very important for fundamental research and practical applications in bio, chemical and medical engineering, especially in designing micro-devices. Recently, micro-devices enable processing, analyzing, and delivering biochemical materials in a wide range of biomedical and biological applications [68, 110]. For example, micro-needle can be used to efficiently and precisely deliver a small amount of drug or DNA into local tissue, skin regions, and even cells. In order to avoid pain and tissue traumas caused by traditional technologies of drug injection and delivery, a variety of micro-needles have been designed for hypodermic injection and transdermal drug delivery [111, 112]. Micro-channels are the main field to deliver and control injected materials. By designing optimal structures of micro-channels or micro-channel networks, it is possible to efficiently control the injection process, either for simple fluids or complex fluids with macromolecules. It is therefore very important to understand the dynamic behavior of macromolecular when passing though micro-channel with different structures.

Recent development of experimental techniques enables us to study the dynamics and rheological properties of macromolecules such as DNA in micro-channels. For example, it

is possible to use fuorescence imaging techniques to visualize the micro-structural conformations of molecules [113]. Optical tweezers have been used to measure the extension properties of single DNA molecules [114]. By using these techniques, some experimental works have been conducted to study the mechanics of macromolecular suspension flows. Perkins et al. in 1995 [115] measured the extension properties of tethered single DNA molecules in a uniform flow. Perkins et al. in 1997 [116] and Smith and Chu in 1998 studied the dynamic behavior of single DNA molecules in an elongation flow [117]. Smith et al. in 1999 [6] observed the dynamic behavior of single DNA molecules in steady shear flows. The flow of molecular suspensions through a micro-channel is more complicated as it is a combination of non-uniform elongation and shear flows. Shrewsbury et al. used epi-fluorescence microscopy to characterize the flow's impact on the conformation of the molecules in microfluidic devices in which the path consists of a large, inlet reservoir connected to a long, rectangular channel followed by a large downstream reservoir [118]. In the device, DNA molecules were observed to undergo elongation, non-uniform shear and compression. Near the channel wall, high shear rates results in dramatic stretching of the molecules, and may also result in chain scission of the macromolecules.

On the other hand, with the development of computational methods and computer hardware, numerical simulations of

the movement and evolution of macromolecules in micro-devices have been more and more popular. Numerical simulation can provide more details on the flow field and conformations of macromolecules by tracking each molecular chain segment. The molecular size of macromolecules is usually in the same order of magnitude as that of the channel and the equivalent Knudsen number is larger or equal to unity [68]. This restricts the applicability of continuum mechanics methods to these flow problems. Molecular dynamics has been used for comparison with worm-like chain [84] and slip length measurements for sheared films [119]. However, but the number of beads in MD simulation is usually small and the time scale is much shorter than the time scale (of the order of second) that for gathering experimental data [6]. In addition, the characteristic size of micro-channels and DNA suspension can range from dozens of nanometers to several micrometers, and even to several millimeters. For mesoscale problems, it is expensive for MD to directly simulate the dynamic behavior of macromolecules in micro-channels.

Compare with molecular dynamics, Monte Carlo relies on statistical mechanics and it generates states according to appropriate Boltzmann probabilities, instead of trying to reproduce the dynamics of a systems. MC can be deal with problems with larger time and space scales than MD, and it has been used to simulate DNA flow through entropic trap array where polymer is modeled by a lattice model with bond fluctuation [120].

As the size of flow field and DNA molecules can be too large to be handled by MD simulation, various mesoscopic methods have been applied in this area. In this area, the Brownian dynamics simulation (BDS) [121,122] is one most common approach. As a simplified version of the Langevin dynamics, Brownian dynamics corresponds to the limit where no average acceleration takes place during the simulation run. Various molecular models have been used to model the DNA molecules, such as the Kramer's bead-rod chain [123], the FENE chain [75] and the worm-like chain in BDS [122]. Among those molecular models in BDS, the worm-like chain is considered to be the most realistic one [8], comparing with experimental measurements. Larson et al. [81] simulated a DNA molecule in an extensional flow, and Hur et al. in shear flow [122]. Except for simulating single molecules, Brownian dynamics simulation has widely been used in simulating the rheological properties of polymer solutions. For example, the simulation of freely-draining flexible polymers in steady linear flows [124], bead-rod chains in start-up of extensional flow [123], and relaxation of dilute polymer solutions following extensional flow [125]. Brownian dynamics simulations have shown a good comparison [122] with experiments on DNA molecules in shear flow. However, these models are usually only valid for simple fluid flow since the flow field has to be specified a-priori in BDS, such as the above mentioned freely-draining flexible polymers in steady linear flow, bead-rod chains in start-up of extensional flow and single DNA molecule in shear flow [122].

Although there are many other mesoscale simulation methods, such as Lattice Gas Automata, Lattice Boltzmann, it is also difficult for those methods to deal with complex fluids and complex flow which may contain macromolecules. In many recent works, dissipative particle dynamics were employed to simulate dynamic behavior of macromolecules. As a mesoscale fluid simulation method, DPD is promising in simulating macromolecules suspension flow through micro channels. Macromolecules (such as DNA molecules) are generally simulated by a series of particles (beads) linked together using springs to mimic the solute molecules, while the simple DPD particles are usually used to model the solvent.

One of the first applications of DPD to modeling dilute polymer solution was presented by Schlijper et al. [86], who applied the DPD method to investigate the link between molecular features of polymer molecules and the rheological properties of dilute polymer solutions. Kong et al investigated the effect of solvent quality (i.e., good solvent vs. poor solvent) on the conformation and relaxation of polymers [126], and the DPD method is applied to simulate single chains in solvents with a range of quality. Groot studied the formation of a polymer–surfactant complex in bulk solution, and DPD is used to model a system containing polymer, surfactant and water [85]. Spenley presented scaling laws for polymer melts and dilute polymer in DPD system [127]. The polymer in a good solvent shows satisfactory agreement with scaling and Kirkwood theory, and the polymer melt is in excellent agreement with the predictions of Rouse theory. Symeonidis et al. [128] demonstrated the correct static scaling laws for the radius of gyration by DPD simulations of several bead-spring representations of polymer chains in dilute solution. They found that the worm-like chain simulating single DNA molecules compares well with average extensions in shear flow from experiments.

Recently, the dynamics of polymer chain in different flows and micro-channels (or micro fluidic devices) were conducted. For example, Wijmans and Smit used DPD to simulate shear flow between two flat plates and to study the effects of shear flow on end-tethered polymer layers ("brushes") [129]. They found that as the flow velocity changes during an oscillation cycle, the polymer chains are able to relax their configurations with respect to the shear rate. Symeonidis modeled λ-phage DNA under shear using DPD with the worm-like chain models [130]. They computed the values of viscosity, diffusivity and Schmidt number and presented comparison of wormlike chain models under shear with experimental and Brownian Dynamics results. Fan et al. [75] simulated the micro channel flows of macromolecular suspension while FENE chain is used in the DPD simulation.
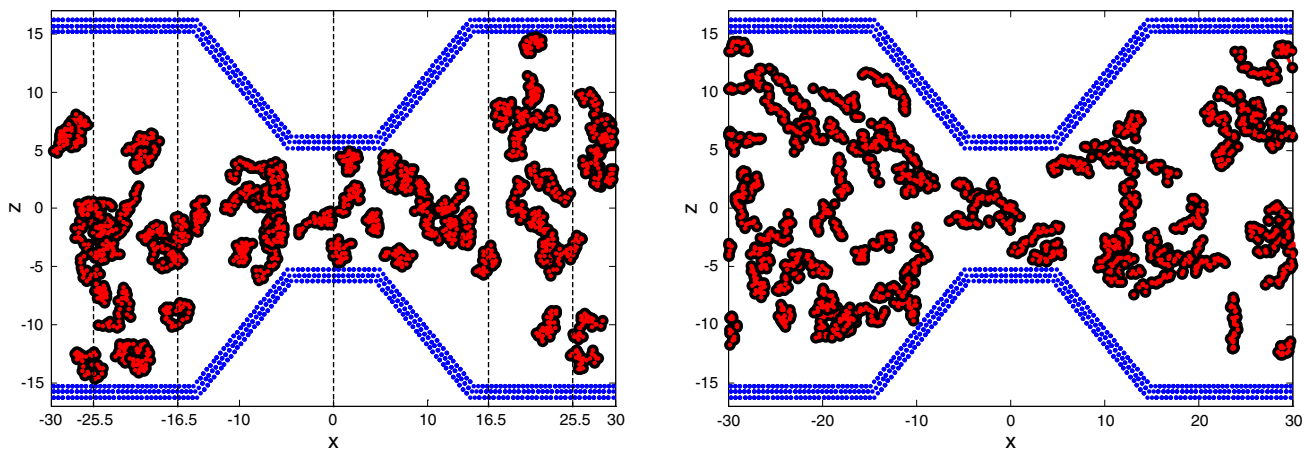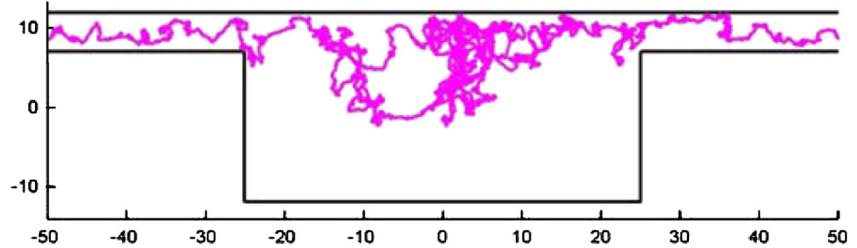
**Fig. 22** Conformation evolution of macromolecules passing through a periodic sloping micro-channel at $t = 0$ (*left*) and 4,000 (*right*) with $nChain = 60$ and $ChainLen = 30$ (from [134])

**Fig. 23** Snapshot of a long DNA molecule in a section of microfluidic sieving channel subjected to an external electric field [57]



They also studied the flow of DNA suspension through periodic contraction–diffusion micro-channels and obtain some typical conformations of macromolecules during evolution of DNA molecules when passing through the channel [68]. Researchers from the same group also conducted other simulations using DPD. For example, Chen et al. [131] investigated the steady-state and transient shear flow dynamics of polymer drops in a micro-channel. Duong-Hong et al. [57] investigated the Electroosmotic flow (EOF) of thin Debye layer and DNA molecular in nano-fluidic systems. Pan et al. [88] simulated DNA separation process through entropic trapping mechanism with worm-like chain. The simulations show that longer DNA strands do move faster than shorter ones, as observed in experiment [132,133]. They confirmed that the delayed entrance is the cause of entropic trapping and concluded that the corner trapping is not a contributor to DNA separation.

Figure 22 shows the conformation evolution of macromolecules passing through a periodic sloping micro-channel at $t = 0$ (left) and 4,000 (right) with $nChain = 60$ (number of macromolecules) and $ChainLen = 30$ (length of a macromolecule) [134]. The macromolecules are initialized in a coiled state. After the flow is fully developed, some macromolecules are in a fully stretched state, while some are still in a coiled state. The extension of the macromolecules is dependent on the slope of the inclined channel. If the slope is very small, the inclined channel will behave somewhat sim-

ilar to a straight channel, and most macromolecules can be fully stretched. If the slope is very big, the inclined channel will behave somewhat similar to square-shaped contraction channel, and it is not easy for the macromolecules to be fully stretched.

One of the excellent features of DPD is the convenience in including additional physics in the model. This section introduces an example of a DPD model considering the Electroosmotic flow (EOF) effects in the small Debye length limit in a DNA sieving device. In addition, the "particle" in this model is a very long NDA molecular chain. The material presented in here is largely from the work by Duong-Hong et al. [57]. It is found that the DPD is and effective coarse-grained model allowing efficient simulation of the hydrodynamics of microfluidic devices of sizes that are too large to be simulated by ab initio methods such as Molecular Dynamics, and too small for continuum models. In this example, EOF is efficiently generated using the proven similitude between velocity and electric field under appropriate conditions. The EOF is generated using an effective boundary condition, akin to a moving wall, thus avoiding evaluation of the computationally expensive electrostatic forces.

A DNA sieving device is a channel consisting of a series of a large number of repetitive shallow and depth sections. The long DNA molecular chain flows through the channel driven by the externally applied electric field. Figure 23 shows a DPD solution of one long DNA molecule position snapshot

in one sieving section. At the two ends, repetitive periotic boundary conditions are used. The numerical data obtained from our model are in very good agreement with theoretical results. The detailed formulation on this interesting DPD model, numerical implementation procedure and the parameters used in the simulation can be found in [57].

## 4.4 Movement and Deformation of Single Cells

Dynamical behaviors of migration and deformation variations of cells in blood vessels are caused by pathological changes in mechanical properties of cells, which may be closely related to severe cell diseases. In cancer, the variations are due to internal factors such as genetic mutation. While in malaria, the variations are due to external factors such as parasites and bioactive lipids. Through biochemical reactions, both internal and external factors can change the internal structure and mechanical behavior of living cells. These variations are often facilitated by the altering in the mechanical behavior of living cells such as large changes of elastic modulus. For example, Healthy red blood cells (RBCs) can squeeze through capillaries with smaller diameter than itself to deliver oxygen to various parts of the body due to its high deformability. While the elastic modulus of RBCs infected by the protozoan Plasmodium may increase by more than a factor of 10 due to intracellular structural changes [135]. The pathological RBCs are too stiff to deform sufficiently to traverse narrow capillaries. Instead they may disrupt the blood flow and block the capillaries, possibly leading to anemia and can even cause death. Modern physiology and medicine have established the relationship of mechanical variations between healthy human cells and pathological cells. For instance, compared to healthy cells, diseased cells such as cancer cells are known to have different stiffness and elasticity [136]. Such differences could be used to distinguish between normal and diseased cells [137,138]. Recently, increased micro-fluidic devices were designed to diagnose and treat cells disease such as cancer as difference cells can have different mechanical properties [139]. It is therefore an important step to understand how cells with different mechanical properties respond to physical loads.

Continuum cell models are main approach to model the mechanic dynamics of cell. The continuum models treat the cell as comprising materials with certain continuum material properties. Appropriate constitutive material models and the associated parameters can derived from experimental observations [140]. Generally, continuum models can be classified into two main categories, namely solid models and liquid drop models. The solid models usually assumed the whole cell to be homogeneous without considering the distinct cortical layer. By assuming homogeneity, the mechanical parameters can be reduced. This greatly simplifies the experimen-

tal data analysis. The corresponding material models can be incompressible elastic solid (linear elastic solid model) or the viscoelastic solid (linear viscoelastic solid model). The homogeneous viscoelastic solid model was first proposed in 1981 [141], which was used to study the human leukocytes undergoing micropipette aspiration with small-strain deformation. For large cell deformations, this model may not work. The elastic model is a simplified version of the viscoelastic model, and it neglects the time factor [140]. This model was used to model endothelial cells in micropipette aspiration [142]. It was found when the pipette radius is very small compared to the local radius of the cell surface, the cell can be approximated as an incompressible elastic half-space. Based on this model, Mijailovich et al. [143] constructed a finite element model to compute cell deformation during magnetic bead twisting experiment. Numerical simulations show that adhesive forces are sufficient to keep the bead firmly attached to the cell surface throughout the range of working torques. Practice shows the solid models can usually achieved equilibrium after certain amount of loading. For instance, even when the suction pressure greatly exceeded the critical suction pressure, endothelial cells and chondrocytes were unable to flow into the pipette [144].

On the other hand, liquid drop models treat the cell as a liquid drop, which can model large cell deformations. The Newtonian liquid drop model was developed by Yeung and Evans in [145] to simulate the flow of cells into the micropipette. The model describes the cell as a homogeneous Newtonian viscous liquid drop enclosed by a cortex with constant, isotropic tension but without bending resistance [146]. Newtonian liquid drop model can model large cell deformations well when the progress of cells in micropipette aspiration is slow. However, for a fast period $<5\,\mathrm{s}$, Newtonian liquid drop model could not explain why the cells would exhibit as a fast elastic recoil, analogous to the initial rapid entry in the aspiration experiment [146]. In order to consider the effects of the nucleus on cell deformation, the compound drop model was developed, which assumed the nucleus as an encapsulated liquid drop [147]. Compare with above mentioned Newtonian liquid drop, the compound drop model can effectively explain the rapid initial response in micro-pipette aspiration and fast recoil on recovery [148]. The compound drop model was also used to model cell under shear flow [149] and extensional flows [150], cell adhesion, and migration [151] as well as shear thinning and membrane elasticity [152]. Recently, Leong et al. presented a modified compound drop model, which can consider stiffness, elasticity, and viscosity of both the cortex and the nucleus to model breast cancer cell entry into a constricted micro-channel. The modeled cell entry behavior agrees with experimental observations [153].

The continuum cell models are easy to implement and straightforward to use in computing the mechanical proper-

ties of the cells if the biomechanical response at the cell level is needed. It provides less insight into the detailed molecular mechanical events. For this reason, more accurate micro- and nano-structural models were developed. However, almost all micro- and nano-structural models were only applicable to red human blood cell. The RBC membrane is composed of a lipid bilayer and an attached cytoskeleton. The cytoskeleton consists primarily of spectrin proteins, which form the network by linking short actin filaments. Discher et al. [154] and Li et al. [155] developed the spectrin-level RBC model. The spectrin-level model corresponds to an effective spectrin network where each spring represents a single spectrin tetramer. The RBC is represented by a network of springs in combination with bending rigidity and constraints for surface-area and volume conservation. The spectrin-level RBC model was successfully validated against experimental data of the mechanical response of an individual cell. However it involves limited degrees of freedom and application of the model in flow simulations requires prohibitively expensive computations. For this reason, Pivkin and Karniadakis developed a coarse-grained model based on the spectrin-level RBC model using mean-field theory and then applied it to DPD simulations in capillaries of $10 \, \mu m$ in diameter while the blood velocity is typically about $1 \, mm/s$ [156]. This RBC, as a collection of DPD particles, is immersed in DPD fluid. The RBC particles interact with the fluid particles through DPD potentials and the temperature of the system is controlled through the DPD thermostat. Initially, the fluid is at rest and the RBC is placed in the middle of the capillary. A body force is applied in the axial direction to drive the flow in the tube. The RBC was found to deform under the flow conditions and, after some transition period, assumed the parachute-type shape, which is commonly observed in experiments [157]. After the body force is turned off, the DPD fluid slows down and eventually returns to rest, while the RBC recovers its equilibrium biconcave shape. A more systematic and rigorous procedure to derive coarse-grained RBC models was present by Fedosov et al. [158]. The RBC is modeled by DPD and captures the elastic response at small and large deformations, which agrees very well with experiments of RBC stretching by optical tweezers. In addition, they also develop a stress-free model which avoids a number of pitfalls of existing RBC models, such as non-smooth or poorly controlled equilibrium shape and dependence of the mechanical properties on the initial triangulation quality. Fedosov et al. [159] also extended this model to model adhesive dynamics of RBCs in Malaria.

Zhou et al. [160] developed a similar approach for modeling other cells in an attempt to simulate the breast cancer cells through a constricted micro-channel. Based on the finite extensible non-linear elastic (FENE) bead spring chain model, they successfully used DPD particles to construct a
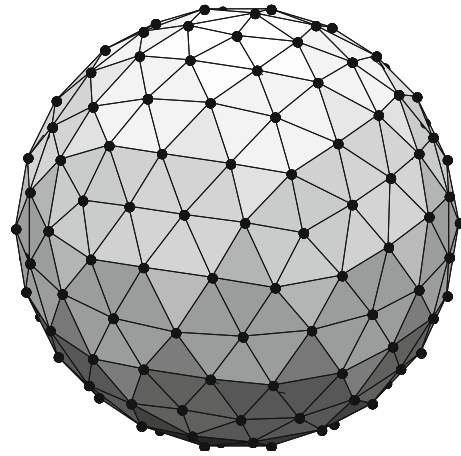


**Fig. 24** Particle model for cell membrane represented by a network of springs linked DPD particles [160]
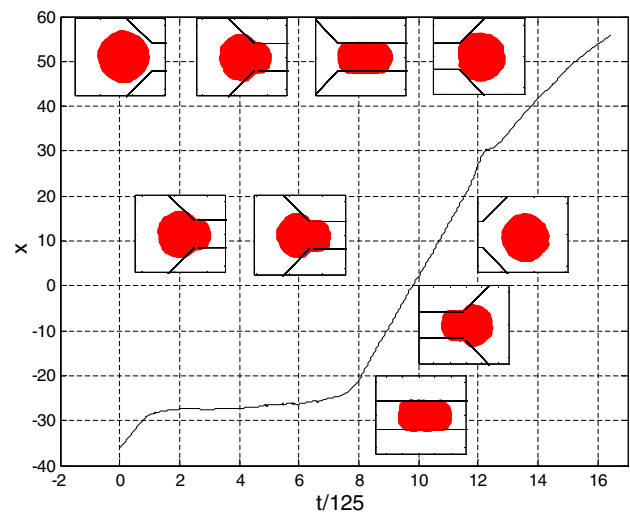


**Fig. 25** Snapshots of a single cell passing through a micro channel, and the distance-to-origin profile. The origin is at the point of entry of cell into the micro-channel [160]

phenomenological breast cancer cell Fig. 24. The model was then used to investigate the transportation and deformation of cell through a constricted micro-channel Fig. 25. It is found that before entering the contracted micro-channel, the cell gradually deforms with reducing velocity. As the cell gradually enters the micro-channel, it elongates until its major axis reaches maximum. As the front of the cell gradually leaves the micro-channel, it restores its original structure with increasing velocity. Those results are agrees with experimental observations [138] and the results obtained from compound drop model [153]. The obtained patterns of cell deformation, contraction and expansion as well the recovery of its original shape are similar to experimental observations. Size effects of the contraction micro-channel and influences of cell membrane properties are also studied.

## 5 Concluding Remarks

Dissipative particle dynamics is a coarse-grained molecular dynamics method, in which a particle contains a cluster of atoms/molecules. As the DPD method is associated with bigger particle size and soft conservative interaction potential, it should be able to model problems with larger time and length scale. As a mesoscale method, DPD is suitable for modeling complex physics and it is able to reproduce continuum hydrodynamic when the time step in the time integration scheme approaches zero and the system size is large enough for hydrodynamic (continuum) concepts and models to be valid. Just as stated by Moeendarbary et al. [9], DPD is "arguably one of the best mesoscale simulation techniques, and in the near future, it has the potential to emerge as an even more widely used modeling and simulation technique for many complex fluid systems".

As a coarse-grained molecular dynamics, DPD has many similarities with MD. Both DPD and MD are deterministic Lagrangian, particle methods (on meso and atomic scales respectively). The computational procedures of DPD and MD are also very much similar including particle initialization, system equilibration, neighbor particle searching, pairwise force computation, physical variable evaluation and even pre- and post-processing. The solid boundary treatment in DPD is also similar to that in MD, and usually involves the use of frozen particles and the inclusion of reflection models.

However, DPD is different from MD in many aspects. One primary feature of DPD is that DPD uses bigger particle and soft conservative interaction potential, and it is therefore can be applied to larger length and time scale than those in MD. This makes the numerical simulation of the movement and suspension of macromolecules such as DNA [with a uncoiled length of $O(10\,\mu m)$] feasible. It is also possible to model the movement and deformation of a single cell [with a diameter of $O(10\mu m)$] or even a cluster of cells possible by using DPD method.

The other main feature of the DPD method is the inclusion of the dissipative force and random force, which act together to maintain an essentially constant temperature with small fluctuations around the nominal temperature. The conservative force in DPD is similar to that in MD, but different in the interaction potential. The interaction potential is important in MD as it determines the pattern of interaction between particles. The interaction potential has been well investigated in MD, and different interaction potentials have been used for different materials/fluids. However, there are very few investigations on the conservative interaction potentials in DPD. The conventional DPD uses a simple conservative force weighting function ($w^C(r) = 1 - r$), which is repulsive and is effective in modeling fluids behaving like gas. It is not able to simulate the flow of liquids with free surfaces, the behavior of bubbly liquids, droplet dynamics and

other important multiphase fluid flow processes. By combining two SPH smoothing functions (of polynomial form), it is possible to construct conservative interaction potentials with short-range repulsion and long-distance attraction. The corresponding strength coefficients and cut-off distances of the repulsive part and attractive part determines the exact form of the interaction potential and further determines the physical properties of the modeling DPD fluid. By using this new interaction potential, it is possible to model complex systems with co-existing liquid–gas–solid phases.

In classic molecular dynamics, dimensional, primal variables can be used in computer implementation, and the modeling parameters can be correlated to the physical properties of real materials/fluids. In contrast, DPD method is generally implemented in a non-dimensional form. In order to match the modeling parameters with the physical properties of real materials/fluids, it should be careful in choosing coefficients in DPD simulation. In general, some coefficients can be determined by fitting the relevant data of the real fluid, some are selected to maintain the numerical accuracy in simulating simple cases with analytical solutions. Another point is that due to the soft interaction between DPD particles, the Schmidt number and dynamic viscosity obtained from a conventional DPD simulation are usually much lower than those for a real fluid. This drawback can be remedied by increasing the cut-off distance or reducing the exponential factor of the weight function of the dissipative (and random) force.

For the modeling of movement and suspension of macromolecules or movement and deformation of single cell, the spring-bead chain models are necessary for simulating the connectivity of neighboring DPD particles on macromolecules or cell. No matter what kind of spring-bead chain model is used, one key point is to match the modeling parameters (such as spring constants) with the physical properties of real materials/fluids. For example, based on an analytical theory, Fedosov developed a DPD model for simulating red blood cell (RBC) to predict RBC mechanics, rheology and dynamics, while the modeled membrane properties can be uniquely related to experimentally determined RBC macroscopic properties [158]. The DPD is also found superior in including additional physics in the model, though adding addition forces. It is also useful for simulating very long particle chains [57].

The computational effort of DPD simulation can be greatly reduced by using proper algorithm and parallel computing techniques. Similar to molecular dynamics simulation, the main computational effort in DPD simulations arises from the calculation of interaction forces between particles. A special neighbor-list algorithm can be used in the DPD code that allows the resultant neighbor-list to remain valid for a number of time steps, typically 10–20 time steps [26]. This neighbor-list algorithm can greatly reduce the compu-

tational time in calculating interaction forces between particles. In addition, the computational performance of the DPD models can be greatly enhanced using parallel computing techniques such as message passing interface (MPI) and Graphic Processing Unit (GPU). The parallelization procedure of a DPD code is very similar to existing parallelization techniques developed for MD and SPH simulations [22].

# References

1. Gardner JW, Varadan VK, Awadelkarim OO (2001) Microsensors, mems, and smart devices. Wiley, New York
2. Hsu T-R (2008) Mems & microsystems: design, manufacture, and nanoscale engineering. Wiley, New York
3. Ho CM, Tai YC (1998) Micro-electro-mechanical-systems (mems) and fluid flows. Annu Rev Fluid Mech 30(1):579–612
4. Karniadakis G, Be k k A, Aluru NR (2005) Microflows and nanoflows: fundamentals and simulation. Springer, New York
5. McAllister DV, Allen MG, Prausnitz MR (2000) Microfabricated microneedles for gene and drug delivery. Annu Rev Biomed Eng 2(1):289–313
6. Smith DE, Babcock HP, Chu S (1999) Single-polymer dynamics in steady shear flow. Science 283(5408):1724–1727
7. Bustamante C, Smith SB, Liphardt J, Smith D (2000) Single-molecule studies of DNA mechanics. Curr Opin Struc Biol 10(3):279–285
8. Bustamante C, Marko JF, Siggia ED, Smith S (1994) Entropic elasticity of lambda-phage DNA. Science 265(5178):1599–1600
9. Moeendarbary E, Ng TY, Zangeneh M (2009) Dissipative particle dynamics: introduction, methodology and complex fluid applications—a review. Int J Appl Mech 1(04):737–763
10. Curtin WA, Miller RE (2003) Atomistic/continuum coupling in computational materials science. Model Simul Mater Sci 11(3):R33
11. Koplik J, Banavar JR (1995) Continuum deductions from molecular hydrodynamics. Annu Rev Fluid Mech 27(1):257–292
12. Zienkiewicz OC, Taylor RL (1977) The finite element method. McGraw-hill, London
13. Hughes TJ (2012) The finite element method: linear static and dynamic finite element analysis. Courier Dover, New York
14. Liu GR, Quek SS (2003) Finite element method: a practical course: a practical course. Butterworth-Heinemann, Oxford
15. Liu GR, Trung NT (2010) Smoothed finite element methods. CRC Press, Boca Raton
16. Chung TJ (2002) Computational fluid dynamics. Cambridge University Press, Cambridge
17. Anderson JD (1995) Computational fluid dynamics: the basics with applications. McGraw-Hill, New York
18. Benson DJ (1992) Computational methods in lagrangian and eulerian hydrocodes. Comput Method Appl Mech Eng 99(2–3):235–394
19. Chow CY (1979) An introduction to computational fluid mechanics. Wiley, New York
20. Liu WK, Chen Y, Jun S, Chen JS, Belytschko T, Pan C, Uras RA, Chang CT (1996) Overview and applications of the reproducing kernel particle methods. Arch Comput Methods Eng 3(1):3–80
21. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P (1996) Meshless methods: an overview and recent developments. Comput Method Appl Mech Eng 139(1–4):3–47
22. Liu GR, Liu MB (2003) Smoothed particle hydrodynamics: a meshfree particle method. World Scientific, Singapore
23. Liu GR (2002) Meshfree methods: moving beyond the finite element method. CRC Press, Boca Raton
24. Onate E, Idelsohn SR, Del Pin F, Aubry R (2004) The particle finite element method. An overview. Int J Comput Methods 1(2):267–307
25. Allen MP, Tildesley DJ (1987) Computer simulation of liquids. Oxford University Press, Oxford
26. Rapaport DC (2004) The art of molecular dynamics simulation. Cambridge University Press, Cambridge
27. Frenkel D, Smit B (2002) Understanding molecular simulation: from algorithms to applications. Academic Press, San Diego
28. Kresse G, Hafner J (1993) Ab initio molecular dynamics for liquid metals. Phys Rev B 47(1):558
29. Kresse G, Hafner J (1994) Ab initio molecular-dynamics simulation of the liquid-metal-amorphous-semiconductor transition in germanium. Phys Rev B 49(20):14251
30. Voter AF (2007) Introduction to the kinetic Monte Carlo method. In: Radiation effects in solids. Springer, New York, pp 1–23
31. Pan LS, Liu GR, Lam KY (1999) Determination of slip coefficient for rarefied gas flows using direct simulation monte carlo. J Micromech Microeng 9(1):89
32. Pan LS, Liu GR, Khoo BC, Song B (2000) A modified direct simulation monte carlo method for low-speed microflows. J Micromech Microeng 10(1):21
33. Ladd AJ (1994) Numerical simulations of particulate suspensions via a discretized boltzmann equation. Part 2. Numerical results. J Fluid Mech 271(1):311–339
34. He XY, Luo LS (1997) Theory of the lattice Boltzmann method: from the boltzmann equation to the lattice Boltzmann equation. Phys Rev E 56(6):6811
35. He XY, Luo LS (1997) A priori derivation of the lattice Boltzmann equation. Phys Rev E 55(6):R6333
36. Luo LS (1998) Unified theory of lattice Boltzmann models for nonideal gases. Phys Rev Lett 81(8):1618
37. Chen SY, Doolen GD (1998) Lattice Boltzmann method for fluid flows. Annu Rev Fluid Mech 30:329–364
38. Pomeau BHY, Frisch U (1986) Lattice-gas automata for the Navier–Stokes equation. Phys Rev Lett 56(14):1505
39. McNamara GR, Zanetti G (1988) Use of the Boltzmann equation to simulate lattice-gas automata. Phys Rev Lett 61(20):2332
40. Lucy LB (1977) A numerical approach to the testing of the fission hypothesis. Astron J 82(12):1013–1024
41. Gingold RA, Monaghan JJ (1977) Smoothed particle hydrodynamics-theory and application to non-spherical stars. Mon Not R Astron Soc 181:375–389
42. Monaghan JJ (1992) Smooth particle hydrodynamics. Annu Rev Astron Astr 30:543–574
43. Cleary PW, Prakash M, Ha J, Stokes N, Scott C (2007) Smooth particle hydrodynamics: status and future potential. Prog Comput Fluid Dy 7(2–4):70–90
44. Xu ZJ, Meakin P, Tartakovsky AM (2009) Diffuse-interface model for smoothed particle hydrodynamics. Phys Rev E 79(3):036702
45. Ren J, Ouyang J, Yang B, Jiang T, Mai H (2011) Simulation of container filling process with two inlets by improved smoothed particle hydrodynamics (sph) method. Int J Comput Fluid Dyn 25(7):365–386
46. Hu X, Adams N (2006) A multi-phase sph method for macroscopic and mesoscopic flows. J Comput Phys 213(2):844–861
47. Tartakovsky AM, Meakin P (2005) A smoothed particle hydrodynamics model for miscible flow in three-dimensional fractures and the two-dimensional rayleigh taylor instability. J Comput Phys 207(2):610–624
48. Aidun CK, Clausen JR (2010) Lattice-Boltzmann method for complex flows. Annu Rev Fluid Mech 42:439–472
49. Monaghan JJ (2005) Smoothed particle hydrodynamics. Rep Prog Phys 68(8):1703–1759

50. Liu MB, Liu GR (2010) Smoothed particle hydrodynamics (sph): an overview and recent developments. Arch Comput Methods Eng 17(1):25–76

51. Koumoutsakos P (2005) Multiscale flow simulations using particles. Annu Rev Fluid Mech 37:457–487

52. Sukop MC, Thorne DT Jr (2007) Lattice boltzmann modeling: an introduction for geoscientists and engineers. Springer, New York

53. Guo ZL, Shu C (2013) Lattice Boltzmann method and its applications in engineering. World Scientific, Singapore

54. Wolf-Gladrow DA (2000) Lattice-gas cellular automata and lattice boltzmann models: an introduction. Springer, New York

55. Hoogerbrugge PJ, Koelman J (1992) Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. Europhys Lett 19:155

56. Yan K, Chen YZ, Han JY, Liu GR, Wang JS, Hadjiconstantinou NG (2012) Dissipative particle dynamics simulation of field-dependent DNA mobility in nanoslits. Microfluid Nanofluid 12(1–4):157–163

57. Duong-Hong D, Wang JS, Liu GR, Chen YZ, Han JY, Hadjiconstantinou NG (2008) Dissipative particle dynamics simulations of electroosmotic flow in nano-fluidic devices. Microfluid Nanofluid 4(3):219–225

58. Wang R, Wang JS, Liu GR, Han JY, Chen YZ (2009) Simulation of DNA electrophoresis in systems of large number of solvent particles by coarsegrained hybrid molecular dynamics approach. J Comput Chem 30(4):505–513

59. Espanol P, Warren P (1995) Statistical mechanics of dissipative particle dynamics. Europhys Lett 30(4):191–196

60. Marsh C (1998) Theoretical aspects of dissipative particle dynamics. University of Oxford, Oxford

61. Groot RD, Warren PB (1997) Dissipative particle dynamics: bridging the gap between atomistic and mesoscopic simulation. J Chem Phys 107(11):4423

62. Pagonabarraga I, Frenkel D (2001) Dissipative particle dynamics for interacting systems. J Chem Phys 115:5015

63. Español P (1997) Fluid particle dynamics: a synthesis of dissipative particle dynamics and smoothed particle dynamics. Europhys Lett 39(6):605

64. Espanol P, Revenga M (2003) Smoothed dissipative particle dynamics. Phys Rev E 67(2):26705

65. Bock H, Gubbins KE, Klapp SHL (2007) Coarse graining of nonbonded degrees of freedom. Phys Rev Lett 98(26):267801. doi:10.1103/PhysRevLett.98.267801

66. Knotts TA IV, Rathore N, Schwartz DC, de Pablo JJ (2007) A coarse grain model for DNA. J Chem Phys 126(8):084901

67. Nielsen SO, Lopez CF, Srinivas G, Klein ML (2004) Coarse grain models and the computer simulation of soft materials. J Phys Condens Mat 16(15):R481

68. Fan XJ, Phan-Thien N, Chen S, Wu XH, Ng TY (2006) Simulating flow of DNA suspension using dissipative particle dynamics. Phys Fluids 18(6):063102. doi:10.1063/1.2206595

69. Pagonabarraga I, Hagen MHJ, Frenkel D (1998) Self-consistent dissipative particle dynamics algorithm. Europhys Lett 42(4):377–382

70. Irving J, Kirkwood JG (1950) The statistical mechanical theory of transport processes. IV. The equations of hydrodynamics. J Chem Phys 18:817

71. Liu MB, Meakin P, Huang H (2006) Dissipative particle dynamics with attractive and repulsive particle-particle interactions. Phys Fluids 18(1):017101. doi:10.1063/1.2163366

72. Liu MB, Meakin P, Huang H (2007) Dissipative particle dynamics simulation of fluid motion through an unsaturated fracture and fracture junction. J Comput Phys 222(1):110–130

73. Revenga M, Zuniga I, Espanol P, Pagonabarraga I (1998) Boundary models in dpd. Int J Mod Phys C 9(08):1319–1328

74. Altenhoff AM, Walther JH, Koumoutsakos P (2007) A stochastic boundary forcing for dissipative particle dynamics. J Comput Phys 225:1125–1136. doi:10.1016/j.jcp.2007.01.015

75. Fan XJ, Phan-Thien N, Yong NT, Wu X, Xu D (2003) Microchannel flow of a macromolecular suspension. Phys Fluids 15(1):11–21. doi:10.1063/1.1522750

76. Liu MB, Meakin P, Huang H (2007) Dissipative particle dynamics simulation of pore-scale flow. Water Resour Res 43:W04411. doi:10.1029/2006WR004856

77. Warren P (2003) Vapour–liquid coexistence in many-body dissipative particle dynamics. arXiv preprint cond-mat/0306027

78. Liu MB, Liu GR, Lam KY (2003) Constructing smoothing functions in smoothed particle hydrodynamics with applications. J Comput Appl Math 155(2):263–284

79. Nugent S, Posch HA (2000) Liquid drops and surface tension with smoothed particle applied mechanics. Phys Rev E 62(4):4968–4975

80. Lebowitz JL, Penrose O (1966) Rigorous treatment of the van der waalsmaxwell theory of the liquidvapor transition. J Math Phys 7:98

81. Larson R, Perkins T, Smith D, Chu S (1999) The hydrodynamics of a DNA molecule in a flow field. In: Flexible polymer chains in elongational flow. Springer, New York, pp 259–282

82. Vologodskii A (1994) DNA extension under the action of an external force. Macromolecules 27(20):5623–5625

83. Warren PB (1998) Dissipative particle dynamics: dynamic aspects of colloids and interfaces. Curr Opin Colloid Interface Sci 3(6):620–624

84. Koelman MVA, Hoogerbrugge PJ (1993) Dynamics simulation of hard-sphere suspensions under steady shear. Europhys Lett 21(3):363–368

85. Groot RD (2000) Mesoscopic simulation of polymer-surfactant aggregation. Langmuir 16(19):7493–7502

86. Schlijper AG, Hoogerbrugge PJ, Manke CW (1995) Computer simulation of dilute polymer solutions with the dissipative particle dynamics method. J Rheol 39:567

87. Groot RD, Rabone KL (2001) Mesoscopic simulation of cell membrane damage, morphology change and rupture by nonionic surfactants. Biophys J 81(2):725–736

88. Pan H, Ng T, Li H, Moeendarbary E (2010) Dissipative particle dynamics simulation of entropic trapping for DNA separation. Sens Actuat A Phys 157(2):328–335

89. Karniadakis GE, Beskok A, Aluru A (2005) Microflows and nanoflows: fundamentals and simulation. Springer, New York

90. Clark AT, Lal M, Ruddock JN, Warren PB (2000) Mesoscopic simulation of drops in gravitational and shear fields. Langmuir 16(15):6342–6350

91. Li Z, Hu GH, Wang ZL, Ma YB, Zhou ZW (2013) Three dimensional flow structures in a moving droplet on substrate: a dissipative particle dynamics study. Phys Fluids 25:072103

92. Zhang MK, Chen S, Shang Z (2012) Numerical simulation of a droplet motion in a grooved microchannel. Acta Phys Sin 61(3):034701. doi:10.7498/aps.61.034701

93. Merabia S, Pagonabarraga I (2006) A mesoscopic model for (de) wetting. Eur Phys J E 20(2):209–214

94. Tiwari A, Abraham J (2006) Dissipative-particle-dynamics model for two-phase flows. Phys Rev E 74(5):056701

95. Boistelle R, Astier J (1988) Crystallization mechanisms in solution. J Cryst Growth 90(1):14–30

96. De Gennes P-G (1985) Wetting: statics and dynamics. Rev Mod Phys 57(3):827

97. Bracke M, De Voeght F, Joos P (1989) The kinetics of wetting: the dynamic contact angle. In: Trends in colloid and interface science iii. Springer, New York, pp 142–149

98. Nativ R, Adar E, Dahan O, Geyh M (1995) Water recharge and solute transport through the vadose zone of fractured chalk under desert conditions. Water Resour Res 31(2):253–261

99. Scanlon BR, Tyler SW, Wierenga PJ (1997) Hydrologic issues in arid, unsaturated systems and implications for contaminant transport. Rev Geophys 35(4):461–490

100. Dragila MI, Weisbrod N (2004) Fluid motion through an unsaturated fracture junction. Water Resour Res 40(2):W02403

101. Kwicklis EM, Healy RW (1993) Numerical investigation of steady liquid water flow in a variably saturated fracture network. Water Resour Res 29(12):4091–4102

102. Persoff P, Pruess K (1995) Twophase flow visualization and relative permeability measurement in natural roughwalled rock fractures. Water Resour Res 31(5):1175–1186

103. Unverdi SO, Tryggvason G (1992) A front-tracking method for viscous, incompressible, multi-fluid flows. J Comput Phys 100(1):25–37

104. Harlow FH (1964) The particle-in-cell computing method for fluid dynamics. Methods Comput Phys 3:319–343

105. Hirt CW, Nichols BD (1981) Volume of fluid (vof) method for the dynamics of free boundaries. J Comput Phys 39(1):201–225

106. Sussman M, Smereka P, Osher S (1994) A level set approach for computing solutions to incompressible two-phase flow. J Comput Phys 114(1):146–159

107. Liu MB, Meakin P, Huang H (2007) Dissipative particle dynamics simulation of multiphase fluid flow in microchannels and microchannel networks. Phys Fluids 19(3):033302. doi:10.1063/1.2717182

108. Cupelli C, Henrich B, Glatzel T, Zengerle R, Moseler M, Santer M (2008) Dynamic capillary wetting studied with dissipative particle dynamics. N J Phys 10(4):043009

109. Huang H, Meakin P, Liu M, McCreery GE (2005) Modeling of multiphase fluid motion in fracture intersections and fracture networks. Geophys Res Lett 32(19):L19402. doi:10.1029/2005GL023899s

110. Chun K, Hashiguchi G, Fujita H (1999) Fabrication of array of hollow microcapillaries used for injection of genetic materials into animal/plant cells. Jpn J Appl Phys 38:279

111. Brazzle JD, Mohanty SK, Frazier AB (1999) Hollow metallic micromachined needles with multiple output ports. In: Symposium on micromachining and microfabrication, international society for optics and photonics

112. Lin L, Pisano AP (1999) Silicon-processed microneedles. J Microelectmech Syst 8(1):78–84

113. Chu S (1991) Laser manipulation of atoms and particles. Science 253(5022):861–866

114. Perkins TT, Quake SR, Smith DE, Chu S (1994) Relaxation of a single DNA molecule observed by optical microscopy. Science 264(5160):822–825 (AAAS-Weekly Paper Edition-including Guide to Scientific Information)

115. Perkins TT, Smith DE, Larson RG, Chu S (1995) Stretching of a single tethered polymer in a uniform flow. Science 268:83–83

116. Perkins TT, Smith DE, Chu S (1997) Single polymer dynamics in an elongational flow. Science 276(5321):2016–2021

117. Smith DE, Chu S (1998) Response of flexible polymers to a sudden elongational flow. Science 281(5381):1335–1340

118. Shrewsbury PJ, Muller SJ, Liepmann D (2001) Effect of flow on complex biological macromolecules in microfluidic devices. Biomed Microdevices 3(3):225–238

119. Cheon M, Chang I, Koplik J, Banavar J (2002) Chain molecule deformation in a uniform flow—a computer experiment. Europhys Lett 58(2):215

120. Tessier F, Labrie J, Slater GW (2002) Electrophoretic separation of long polyelectrolytes in submolecular-size constrictions: a Monte Carlo study. Macromolecules 35(12):4791–4800

121. Northrup SH, Allison SA, McCammon JA (1984) Brownian dynamics simulation of diffusioninfluenced bimolecular reactions. J Chem Phys 80:1517

122. Hur JS, Shaqfeh ES, Larson RG (2000) Brownian dynamics simulations of single DNA molecules in shear flow. J Rheol 44:713

123. Doyle PS, Shaqfeh ES (1998) Dynamic simulation of freely-draining, flexible bead-rod chains: start-up of extensional and shear flow. J Non-Newton Fluid 76(1):43–78

124. Doyle PS, Shaqfeh ES, Gast AP (1997) Dynamic simulation of freely draining flexible polymers in steady linear flows. J Fluid Mech 334:251–291

125. Doyle PS, Shaqfeh ES, McKinley GH, Spiegelberg SH (1998) Relaxation of dilute polymer solutions following extensional flow. J Non-Newton Fluid 76(1):79–110

126. Kong Y, Manke C, Madden W, Schlijper A (1997) Effect of solvent quality on the conformation and relaxation of polymers via dissipative particle dynamics. J Chem Phys 107:592

127. Spenley N (2000) Scaling laws for polymers in dissipative particle dynamics. Europhys Lett 49(4):534

128. Symeonidis V, Karniadakis GE, Caswell B (2005) Dissipative particle dynamics simulations of polymer chains: Scaling laws and shearing response compared to DNA experiments. Phys Rev Lett 95(7):076001

129. Wijmans C, Smit B (2002) Simulating tethered polymer layers in shear flow with the dissipative particle dynamics technique. Macromolecules 35(18):7138–7148

130. Symeonidis V, Karniadakis G, Caswell B (2005) Simulation of λ-phage DNA in microchannels using dissipative particle dynamics. Tech Sci 53(4):395–403

131. Chen S, Phan-Thien N, Fan XJ, Khoo BC (2004) Dissipative particle dynamics simulation of polymer drops in a periodic shear flow. J Non-Newton Fluid 118(1):65–81

132. Han J, Turner S, Craighead H (1999) Entropic trapping and escape of long DNA molecules at submicron size constriction. Phys Rev Lett 83(8):1688

133. Han J, Craighead H (2000) Separation of long DNA molecules in a microfabricated entropic trap array. Science 288(5468):1026–1029

134. Zhou LV, Liu MB, Chang JZ (2012) Dissipative particle dynamics simulations of macromolecules in micro-channels. Acta Polym Sin 7:720–727

135. Suresh S (2006) Mechanical response of human red blood cells in health and disease: some structure-property-function relationships. J Mater Res 21(08):1871–1877

136. Lee GY, Lim CT (2007) Biomechanics approaches to studying human diseases. Trends Biotechnol 25(3):111–118

137. Bathe M, Shirai A, Doerschuk CM, Kamm RD (2002) Neutrophil transit times through pulmonary capillaries: the effects of capillary geometry and fMLP-stimulation. Biophys J 83(4):1917–1933

138. Hou H, Li Q, Lee G, Kumar A, Ong C, Lim C (2009) Deformability study of breast cancer cells using microfluidics. Biomed Microdevices 11(3):557–564

139. Suresh S (2007) Biomechanics and biophysics of cancer cells. Acta Mater 55(12):3989–4014

140. Lim C, Zhou E, Quek S (2006) Mechanical models for living cells—a review. J Biomech 39(2):195–216

141. Schmid-Schönbein G, Sung K, Tözeren H, Skalak R, Chien S (1981) Passive mechanical properties of human leukocytes. Biophys J 36(1):243–256

142. Theret DP, Levesque M, Sato M, Nerem R, Wheeler L (1988) The application of a homogeneous half-space model in the analysis of endothelial cell micropipette measurements. J Biomech Eng 110(3):190–199

143. Mijailovich SM, Kojic M, Zivkovic M, Fabry B, Fredberg JJ (2002) A finite element model of cell deformation during magnetic bead twisting. J Appl Physiol 93(4):1429–1436

144. Jones WR, Ping Ting-Beall H, Lee GM, Kelley SS, Hochmuth RM, Guilak F (1999) Alterations in the young's modulus and volumetric properties of chondrocytes isolated from normal and osteoarthritic human cartilage. J Biomech 32(2):119–127

145. Yeung A, Evans E (1989) Cortical shell-liquid core model for passive flow of liquid-like spherical cells into micropipets. Biophys J 56(1):139–149

146. Evans E, Yeung A (1989) Apparent viscosity and cortical tension of blood granulocytes determined by micropipet aspiration. Biophys J 56(1):151–160

147. Hochmuth R, Ting-Beall H, Beaty B, Needham D, Tran-Son-Tay R (1993) Viscosity of passive human neutrophils undergoing small deformations. Biophys J 64(5):1596–1601

148. Tran-Son-Tay R, Kan H-C, Udaykumar H, Damay E, Shyy W (1998) Rheological modelling of leukocytes. Med Biol Eng Comput 36(2):246–250

149. Agresar G, Linderman J, Tryggvason G, Powell K (1998) An adaptive, cartesian, front-tracking method for the motion, deformation and adhesion of circulating cells. J Comput Phys 143(2):346–380

150. Kan H-C, Shyy W, Udaykumar H, Vigneron P, Tran-Son-Tay R (1999) Effects of nucleus on leukocyte recovery. Ann Biomed Eng 27(5):648–655

151. N'dri N, Shyy W, Tran-Son-Tay R (2003) Computational modeling of cell adhesion and movement using a continuum-kinetics approach. Biophys J 85(4):2273–2286

152. Marella SV, Udaykumar H (2004) Computational analysis of the deformability of leukocytes modeled with viscous and elastic structural components. Phys Fluids 16:244

153. Leong FY, Li Q, Lim CT, Chiam K-H (2011) Modeling cell entry into a micro-channel. Biomech Model Mechanobiol 10(5):755–766

154. Discher DE, Boal DH, Boey SK (1998) Simulations of the erythrocyte cytoskeleton at large deformation. II. Micropipette aspiration. Biophys J 75(3):1584–1597

155. Li J, Dao M, Lim C, Suresh S (2005) Spectrin-level modeling of the cytoskeleton and optical tweezers stretching of the erythrocyte. Biophys J 88(5):3707–3719

156. Pivkin IV, Karniadakis GE (2008) Accurate coarse-grained modeling of red blood cells. Phys Rev Lett 101(11):118105

157. Tomaiuolo G, Preziosi V, Simeone M, Guido S, Ciancia R, Martineelli V, Rinaldi C, Rotoli B (2007) A methodology to study the deformability of red blood cells flowing in microcapillaries in vitro. Ann 'Ist Super sanità 43(2):186–192

158. Fedosov DA, Caswell B, Karniadakis GE (2010) Systematic coarse-graining of spectrin-level red blood cell models. Comput Method Appl Mech Eng 199(29):1937–1948

159. Fedosov DA, Caswell B, Karniadakis GE (2011) Wall shear stress-based model for adhesive dynamics of red blood cells in malaria. Biophys J 100(9):2084–2093

160. Zhou LV, Liu MB, Chang JZ (2012) Dissipative particle dynamics simulations of cell micropipetting. Adances in computational mechanics of granular materials. Dalian University of Technology Press, Dalian