

B 5 Mesoscale hydrodynamics simulations

M. Ripoll

Institut für Festkörperforschung

Forschungszentrum Jülich GmbH

Contents

1	Introduction	2
1.1	Brownian dynamics (BD)	3
1.2	Smoothed particle hydrodynamics (SPH)	4
1.3	Direct simulation Monte Carlo (DSMC)	4
1.4	Lattice gas automata (LGA)	4
2	Lattice Boltzmann (LB)	5
2.1	Boltzmann equation	5
2.2	Lattice Boltzmann method	6
2.3	Further considerations	10
3	Dissipative particle dynamics (DPD)	11
3.1	Isothermal DPD fluid	11
3.2	Applications	16
3.3	Energy conserving DPD model (DPD+e)	17
4	Multiparticle collision dynamics (MPC)	19
4.1	MPC fluid model	20
4.2	Transport properties	24
4.3	Implementation of complex structures	25

1 Introduction

Macroscopic descriptions of fluid properties have been extensively studied on the basis of the work of Navier and Stokes from the first half of the 19th century. The Navier-Stokes equation (see Chapter B1) accounts for the global and local conservation of momentum. The complete description of the fluid requires also the continuity equation for the mass conservation and the heat equation for the energy conservation. In a few cases these equations can be solved analytically, but in general fluid behavior predictions require to solve coupled sets of nonlinear partial differential field equations by use of finite difference or finite element methods. This is the basis for a well established set of methods which is still widely employed. Nevertheless, there are many situations where these methods can not be applied. This can be due to the no applicability of the continuum equations, to the difficulties in obtaining numerical convergence or to the lack of thermal fluctuations. Relevant examples of such limitations appear in problems in which the microscopic properties of the fluid components are important in determining the overall fluid behavior.

From a different perspective, the previous chapters have been devoted to simulation methods based on a microscopic description of fluids, like Molecular dynamics simulation (MD). These techniques would reproduce the exact dynamical, equilibrium or steady state properties of the complex fluids if unlimited computing power would be available. For example, colloidal suspensions are dispersions of solid particles whose sizes range into those defined as mesoscopic scales ($10nm - 1\mu m$). These particles are composed themselves of a large number of atoms (10^6 - 10^9) and the same amount or one order of magnitude higher would be required for the surrounding solvent. To simulate in all detail suspensions of a certain number of colloids with the surrounding solvent and reach a relevant time scale is, therefore, far away from our computational possibilities. However, these intermediate scales (see Fig. 1) are essential to understand a large number of macroscopic phenomena. Most soft matter systems fall into this range of mesoscopic systems ranging from polymers, colloids, droplets, liquid crystals, mixtures, porous media and many biological matter like membranes, or vesicles. Furthermore, many are the processes in which these systems display interesting features like shear flow, microfluidics, microphase separation, or membrane structuring among others. The understanding of these phenomena is still an outstanding challenge for our basic knowledge of fluid mechanics, and moreover, have a considerable practical relevance.

The aim to 'bridge the length and time scales gap', and the increasing availability of computing power, has stimulated the development of several mesoscale simulation techniques in recent years. This has been done fundamentally either from the 'top-down' approach which consists on discretizations of the continuum equations, or from the 'bottom-up' approach which consist in a coarse grained description of the fluid where the microscopic scale is strongly simplified, but relevant effects are still taken into account.

In this line, a first mesoscopic model of colloids would be to treat them as hard spheres, such that the whole structure is summarized as impenetrable objects. This has indeed been successful in predicting static properties of colloidal dispersions in quite early simulations. However, in order to describe dynamical properties of colloidal dispersions, one should also consider solvent effects, and the detail in which this has to be done will depend on the particular effect that one wants to reproduce.

In this chapter, I will first briefly mention most of the existing mesoscopic hydrodynamic simulation techniques as they have historically appeared, giving then a closer description of the basic implementation details of the three methods which could be considered the most competi-

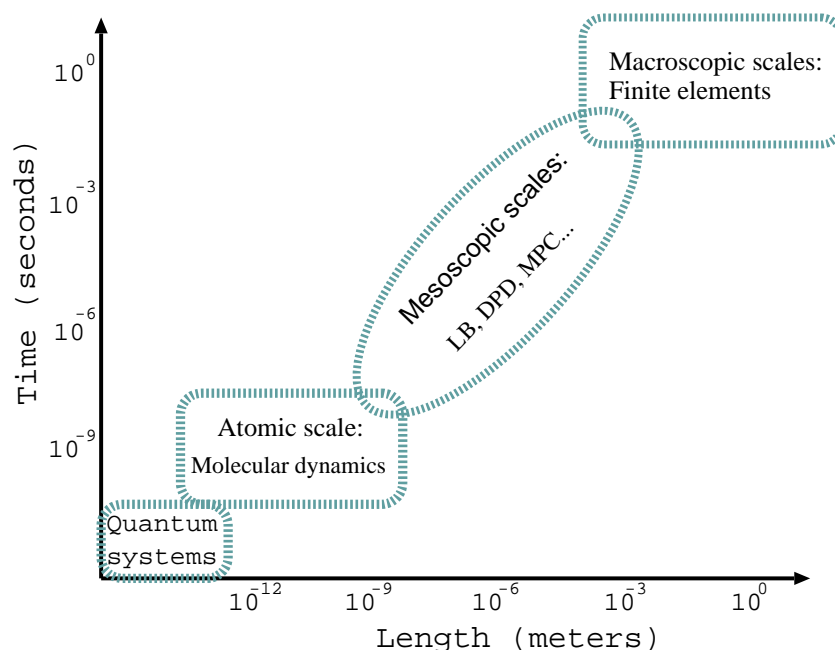


Fig. 1: *Mesoscale dynamics bridges microscopic and macroscopic scales.*

tive ones nowadays, namely Lattice Boltzmann (LB), Dissipative particle dynamics (DPD) and Multiparticle collision dynamics (MPC). Note however, that this chapter is only intended as a first approach to the emerging field of mesoscopic hydrodynamic simulation techniques, and it only includes the basic implementation steps, since a detailed explanation of all the relevant extensions existing in literature would by far exceed the scope of this lecture. On the other hand, the selection of the aspects that will be discussed has unavoidably been biased by my own experience in the field. For further explanations, I refer the interested reader to more extended reviews [1–4] and to the original literature.

1.1 Brownian dynamics (BD)

In BD the solvent particles are omitted from the simulation [5], and their effects upon the solute are represented by a combination of random forces and frictional terms (see Chapter B4). The BD method computes the space trajectories of a collection of particles that individually obey Langevin equations in a field of force, which reproduces the diffusive behavior of for instance colloidal dispersions. The main drawback of the method is that momentum is not a locally conserved quantity, which means that the behavior is not hydrodynamic but diffusive. Hydrodynamic interactions between particles can be incorporated through a tensorial dependence in the Langevin equations. The most common choice for such a dependence is the hydrodynamic Oseen tensor (see Chapter B1). This method has reproduced the behavior of polymer and colloidal dispersions, but the two main problems are that detailed time dependent information about the solvent is lost, and that the size of the considered tensor increases with the number of particles, what limits dramatically the size of the systems that can be considered (see more extensive introduction of this method in Chapter B4).

1.2 Smoothed particle hydrodynamics (SPH)

This technique was developed in the 1970s in the context of astrophysical flow problems [6, 7] and more recently it has been applied to the study of fluid dynamic problems like viscous [8] and thermal flows [9] in simple geometries. This new application is frequently named Smoothed Particle Applied Mechanics (SPAM). The essence is to discretize the macroscopic partial differential equations, such as the Navier-Stokes equations for Newtonian fluids or the elasticity equation for solids. The discretization takes place in an irregular and Lagrangian moving grid in such a way that the nodes can be interpreted as soft particles. In fact, the technique allows one to solve partial differential equations with molecular dynamic simulation codes. SPH has not been applied to study hydrodynamic problems where fluctuations are relevant, as those occurring in colloidal suspensions. Actually, there are some subtleties in discretizing the random stress and random heat flux that appear in the equations of fluctuating hydrodynamics of Landau and Lifschitz [10].

1.3 Direct simulation Monte Carlo (DSMC)

Developed for simulating flow of relatively dilute gases [11], DSMC consists in a set of particles that alternate streaming and collision steps. In the streaming step all particles move ballistically, and then collisions are performed among randomly selected pairs of particles. The space is divided into collision boxes inside which the mentioned pairs are chosen. The number of collisions per time step is fixed by the known collision frequency at the specified density. The precise rules for the collisions depend on the molecular model that one wants to reproduce. The only requirement for these collisions is that mass, linear momentum, and energy are conserved quantities. An extension of the DSMC for dense gases has been recently proposed [12].

1.4 Lattice gas automata (LGA)

In lattice gas automata [13, 14] the continuum macroscopic picture is replaced by a set of particles that move from site to site on a fixed regular lattice. During each time step all the particles move simultaneously according to their current momentum vector. If two particles happen to end up on the same lattice site, they collide and change their velocities according to certain collision rules, whose only restriction is that particle number, momentum and energy should be conserved quantities.

This system, provided an adequate symmetry of the lattice, is a valid fluid-dynamical model yielding the correct Navier-Stokes hydrodynamics at a coarse-grained scale. A noticeable advantage is that any lattice node can be marked as solid, what allows the integration of arbitrarily complex geometries that would be difficult to model with conventional continuum methods due to convergence problems. Comparing with a purely microscopic description, the number of elementary operations per particle and per time step is smaller for LGA than for MD as a consequence of the discretization. Moreover, the time intervals simulated with LGA are orders of magnitude larger: in a single LGA time step changes in the relative particle positions are typically comparable to the mean free path of the particles, whereas in MD the corresponding changes in local particle configurations usually require hundreds of time steps. Examples of complex fluid systems characterized by a multiplicity of length scales that have been simulated with success using LGA are colloidal suspensions [15] or polymer solutions [16].

However, fundamental problems are displayed by this technique, isotropy and Galilean invariance are both broken by the lattice and large density fluctuations appear. Anisotropy in the flow behavior can be eliminated by choosing a lattice with sufficient rotational symmetry. The consequences of the lack of Galilean invariance for the flow behavior, such as unphysical advection terms, can be removed by rescaling the velocities. The occurrence of large fluctuations can be smeared out by local averaging procedures, where a group of neighboring vectors are summarized into a coarse grained vector. However, for dispersed systems and other fluid systems which are more complex than single-phase fluids, these problems show up in a much more severe form. When going to more complicated fluid systems the model becomes quite cumbersome. The appealing initial simplicity is lost and further progress is difficult.

2 Lattice Boltzmann (LB)

The Lattice Boltzmann Equation was first developed empirically [17] from LGA and it was introduced to circumvent two of its major shortcomings: intrinsic noise and limited values of transport coefficients. Later it has been demonstrated that the LBE can be directly derived from the continuous Boltzmann equation [18].

The main purpose is to incorporate the physical nature of fluids from a more statistical point of view. Particle density distribution functions are used instead of single particles, according to the underlying picture of the Boltzmann transport equation, and the dynamics is implemented directly to the distribution functions instead on the individual particles. For a more detailed review of the method see [2, 19, 20].

In the following sections, I will introduce first a brief introduction to the Boltzmann equation with the standard approximations employed for its discretization. Second, I will give the basic concepts for the implementation of the method.

2.1 Boltzmann equation

The primary variable of interest is the one-particle distribution function $f(\mathbf{r}, \mathbf{v}, t)$ (see Chapter B1) which is the probability of one particle to be at time t , at the position \mathbf{r} , with velocity \mathbf{v} . The general time evolution of this distribution is

$$(\partial_t + \mathbf{v} \cdot \nabla) f(\mathbf{r}, \mathbf{v}, t) = (\partial_t f)_{coll} \quad (1)$$

in the absence of external forces. In order to define the Boltzmann equation, the collision operator $(\partial_t f)_{coll}$ has to be explicitly specified. Two fundamental assumptions need to be made. One is the *molecular chaos assumption* (velocity and position of a molecule are uncorrelated quantities), and the second one is that only binary collisions are taken into account (valid in the case of a dilute gas). A further simplification, known as the *collision interval theory*, can be made for states close to equilibrium. Given the relaxation time τ , the collision interval theory assumes that during a time interval h a fraction h/τ of the particles in a small volume collide making the one particle distribution function change from its instantaneous value to the equilibrium one. The equilibrium state is given by the Maxwell-Boltzmann distribution

$$f^{eq} = \frac{\rho}{(2\pi k_B T)^{d/2}} \exp \left[-\frac{(\mathbf{v} - \mathbf{u})^2}{2k_B T} \right], \quad (2)$$

where d is the space dimension, k_B is the Boltzmann constant, T the temperature, ρ the density and \mathbf{u} the macroscopic velocity. With these considerations, the collision term can be expressed in the form known as the 'BGK collision operator' (Bhatnager-Gross-Krook) ,

$$(\partial_t f)_{coll} = -\frac{f - f^{eq}}{\tau}. \quad (3)$$

The Boltzmann equation in Eq. (1) is then,

$$(\partial_t + \mathbf{v} \cdot \nabla) f = -\frac{f - f^{eq}}{\tau}. \quad (4)$$

In order to discretize the differential operators in the l.h.s. of this equation, the original form of LB made the following choice

$$f(\mathbf{r} + \mathbf{v}h, t + h) - f(\mathbf{r}, t) = -\frac{f(\mathbf{r}, t) - f^{eq}(\mathbf{r}, t)}{\tau}h, \quad (5)$$

where h is a discrete time step.

2.2 Lattice Boltzmann method

The Lattice Boltzmann method was historically developed from LGA in an empiric manner. Therefore, similar to LGA, LB consists in a set of particles that move in a space restricted to the nodes of a regular lattice. Particle distributions propagate from node to node, with a rate proportional to a discrete velocity \mathbf{c}_k and interchange mass and momentum with other particle distributions in the corresponding node before the next propagation step. In fact, the LB method corresponds to a formal discretization in the phase space of the Boltzmann equation with $f \rightarrow f_k$ and $\mathbf{v} \rightarrow \mathbf{c}_k$ in Eq. (5).

Typical mesh types for LB - Lattice Boltzmann models are spatially discrete approaches to fluid dynamics. This means that the underlying grids of such simulations must fulfill certain symmetry conditions in order to recover hydrodynamic behavior with full rotational symmetry of space. The most frequent mesh types are the $D1Q3$, $D2Q9$, $D3Q15$ and $D3Q19$ -lattice (see two of them in Fig. 2), where the terminology $DdQn$ refers to the space dimensionality d and to the discrete number n of velocity vectors \mathbf{c}_k which will constitute the vector basis of the distribution function. Lattices are composed of r sublattices defined by velocity vectors with the same velocity magnitude, the most common ones are 0, 1, $\sqrt{2}$, and $\sqrt{3}$.

The $D1Q3$ -lattice has 2 sublattices and 3 velocity vectors (identity, and 2 to the sides). The $D2Q9$ -lattice has 3 sublattices and 9 discrete velocities vectors (identity, 4 towards the face centers and 4 towards vertices of a square). The $D3Q15$ -lattice has 3 sublattices and 15 discrete velocities vectors (identity, 6 towards the face centers and 8 towards vertices of a cube). The $D3Q19$ -lattice has 3 sublattices and 19 discrete velocities vectors (identity, 6 towards the face centers and 12 towards the edge centers of a cube). While the $D3Q15$ -lattice requires less computation and less memory than the $D3Q19$ -lattice, it suffers more from finite size effects and it is less accurate.

For more clarity, I will focus on the $D2Q9$ -lattice (see Fig. 2), whose velocity vectors are

$$\begin{aligned} \mathbf{c}_0 &= (0, 0); & \mathbf{c}_1 &= (0, 1); & \mathbf{c}_2 &= (1, 0); & \mathbf{c}_3 &= (-1, 0); & \mathbf{c}_4 &= (0, -1); \\ & & \mathbf{c}_5 &= (1, 1); & \mathbf{c}_6 &= (-1, 1); & \mathbf{c}_7 &= (-1, -1); & \mathbf{c}_8 &= (1, -1) \end{aligned} \quad (6)$$

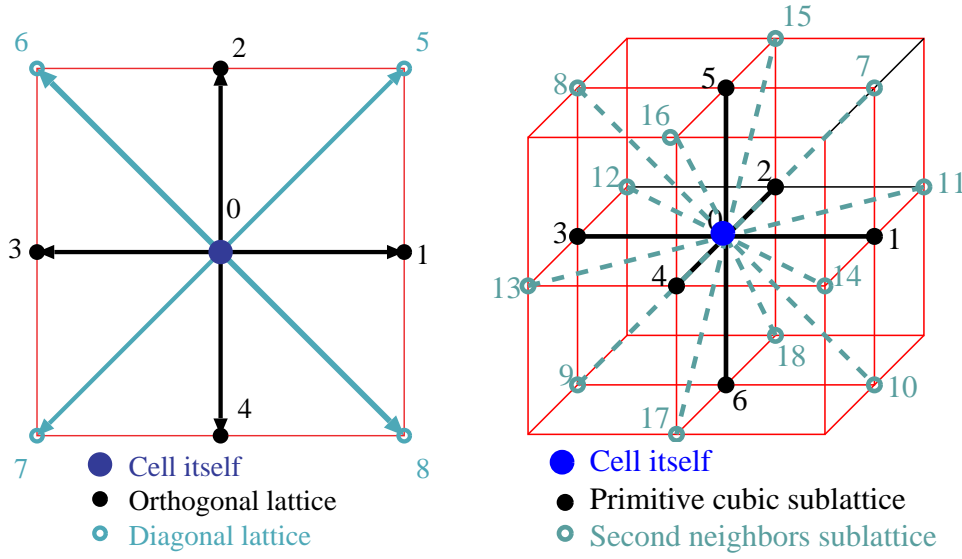


Fig. 2: LB velocity vectors of the D2Q9 (left) and D3Q19 lattice geometry (right).

LB algorithm - A set of one particle distribution functions $f_k(\mathbf{x}_i, t)$ is defined in each lattice node i with k denoting to the n velocities of the corresponding $DdQn$ -lattice. Each of these f_k can be interpreted as the density of fluid that will move in direction k at time t . Since the LB method corresponds to the discretization of the Boltzmann equation in the phase space, the evolution of the densities in each time step can be written from Eq. (5) as,

$$f_k(\mathbf{x}_i + \mathbf{c}_k, t + 1) = f_k(\mathbf{x}_i, t) + \frac{1}{\tau} [f_k^{eq}(\mathbf{x}_i, t) - f_k(\mathbf{x}_i, t)], \quad (7)$$

with the time step made one for simplicity. In numerical applications the above evolution equation is usually split into two subroutines, a collision and a streaming step. In the collision step the set of densities at a given node are relaxed towards equilibrium

$$\tilde{f}_k(\mathbf{x}_i, t) = f_k(\mathbf{x}_i, t) + \frac{1}{\tau} [f_k^{eq}(\mathbf{x}_i, t) - f_k(\mathbf{x}_i, t)]. \quad (8)$$

This relaxation is performed ensuring mass and momentum conservation at each node. The energy is usually not conserved (although energy conserving schemes also exist). In the streaming step densities are moved along their corresponding velocity vectors to the appropriate neighboring site,

$$f_k(\mathbf{x}_i + \mathbf{c}_k, t + 1) = \tilde{f}_k(\mathbf{x}_i, t). \quad (9)$$

Given the structure of a particular lattice, local mass and momentum densities can be calculated in each lattice site as,

$$\begin{aligned} \rho(\mathbf{x}_i, t) &= \sum_k f_k(\mathbf{x}_i, t), \\ \rho(\mathbf{x}_i, t)\mathbf{u}(\mathbf{x}_i, t) &= \sum_k f_k(\mathbf{x}_i, t)\mathbf{c}_k. \end{aligned} \quad (10)$$

Lattice units - It should be noted, that in the previous expressions all the quantities have been chosen to be dimensionless. This corresponds to the most common description of the LB model in which *lattice units* correspond to,

Mass unit:	m	\equiv particle mass	$\rightarrow m f_k$
Length unit:	a	\equiv lattice constant	$\rightarrow a x_i$
Time unit:	h	\equiv time between collisions	$\rightarrow h t, h \tau$

where the last column corresponds to the dimensionalized version of the elementary variables in Eqs. (7)-(10), and similarly for composed quantities like the velocity $(a/h)c_k$, the mass density $\varrho(\mathbf{x}_i a, t h) = \frac{m}{a^3} \rho(\mathbf{x}_i a, t h)$ or the momentum density $\mathbf{j}(\mathbf{x}_i a, t h) = \frac{m}{a^2 h} \rho(\mathbf{x}_i a, t h) \mathbf{u}(\mathbf{x}_i a, t h)$. In the following I will adopt the dimensionless notation in lattice units.

Relaxation time, τ - For incompressible isothermal materials, the relaxation time τ is a parameter which quantifies the rate of change towards local equilibrium. With the BGK relaxation in Eq.(7) all particle distribution functions relax at the same rate, $\omega = 1/\tau$, towards their corresponding equilibrium value. For the method to be stable [21], and for the particle density and viscosity to be positive, the relaxation rate must obey $0 < \omega < 2$. The condition where $0 < \omega < 1$ is called the sub-relaxation regime, while $1 < \omega < 2$ is referred to as over-relaxation regime.

Furthermore, the relaxation time characterizes the behavior of the simulated material since, in the hydrodynamic limit of this equation, it is related with the macroscopic kinematic viscosity ν according to

$$\nu = \frac{(2\tau - 1)c_s^2}{2}, \quad (11)$$

for incompressible isothermal flows, where c_s is the sound speed which depends on the lattice. For $D2Q9$ and $D3Q19$, this is $c_s = 1/\sqrt{3}$.

Determining the equilibrium distribution - The choice of the equilibrium densities $f_k^{eq}(\mathbf{x}_i, t)$ is one of the key ingredients of the model and they are determined by enforcing that the collision operator conserves mass and momentum. The zeroth and first moments of the equilibrium distribution function should then take the form,

$$\begin{aligned} \rho(\mathbf{x}_i, t) &= \sum_k f_k^{eq}(\mathbf{x}_i, t), \\ \rho(\mathbf{x}_i, t) \mathbf{u}(\mathbf{x}_i, t) &= \sum_k f_k^{eq}(\mathbf{x}_i, t) \mathbf{c}_k. \end{aligned} \quad (12)$$

Note that these distributions are the same as those in Eq. (10). It follows then immediately from the evolution equation Eq.(7) that density and momentum are locally conserved.

A low velocity expansion of the Maxwellian distribution in Eq. (2) for constant temperature can be calculated as,

$$f^{eq} \simeq \frac{\rho}{(2\pi k_B T)^{d/2}} \exp \left[-\frac{v^2}{2k_B T} \right] \times \left\{ 1 + \frac{\mathbf{v} \cdot \mathbf{u}}{k_B T} + \frac{(\mathbf{v} \cdot \mathbf{u})^2}{2(k_B T)^2} - \frac{\mathbf{u}^2}{k_B T} \right\} + \mathcal{O}(u^3). \quad (13)$$

Based on this expansion, a suitable form for the discretized equilibrium distribution function is a quadratic function in velocities,

$$f_k^{eq} = A_a + B_a \mathbf{c}_k \cdot \mathbf{u} + C_a u^2 + D_a [\mathbf{c}_k \cdot \mathbf{u}]^2, \quad (14)$$

where a labels the different sublattices, A_a , B_a , C_a , and D_a are coefficients calculated for every lattice geometry, such that symmetry and conservation requirements are satisfied.

As an example, I illustrate now how to calculate the coefficients in Eq. (14) with the restrictions in Eq.(12) in the case of a $D2Q9$ lattice. Using the velocity vectors in Eq.(6) and the general 2-d velocity vector $\mathbf{u} = (u_x, u_y)$, one finds

$$\left. \begin{aligned} f_0^{eq} &= A_0 + C_0 u^2 \\ \sum_{k=1,4} f_k^{eq} &= 4A_1 + 4C_1 u^2 + 2D_1 u^2 \\ \sum_{k=5,8} f_k^{eq} &= 4A_2 + 4C_2 u^2 + 4D_2 u^2 \end{aligned} \right\} \Rightarrow \rho \equiv \begin{aligned} &A_0 + 4A_1 + 4A_2 \\ &+ u^2(C_0 + 4C_1 + 4C_2 + 2D_1 + 4D_2), \end{aligned}$$

$$\sum_{k=0,8} f_k^{eq} \mathbf{c}_k = (2B_1 + 4B_2) \mathbf{u} \equiv \rho \mathbf{u}. \quad (15)$$

A non-trivial set of parameters that satisfies the previous relations is

$$\begin{aligned} A_0 &= \rho/2, & B_1 &= \rho/4, & C_0 &= -3\rho/4, & D_1 &= \rho/2, \\ A_1 &= \rho/12, & B_2 &= \rho/8, & C_1 &= -\rho/8, & D_2 &= \rho/8. \end{aligned} \quad (16)$$

$$A_2 = \rho/24, \quad C_2 = -\rho/16,$$

More complicated lattices symmetries follow the same procedure and further conservation laws can be performed as well.

Implementation of walls - For many situations, it is necessary to implement of walls as boundary conditions. Two main types of boundary conditions are possible when simulating a solid wall, these are stick and slip. This classification is performed attending the difference of velocity between the fluid and the wall. *Slip boundary conditions* allow that the fluid in contact with the wall and the wall itself have different velocities. *Stick boundary conditions* (also called no-slip) impose a continuous change of velocity between the wall and the fluid. The standard procedure to obtain stick is known as *bounce-back*. When a particle hits the wall its trajectory is reverted and its velocity is inverted. These considerations apply to many simulation techniques.

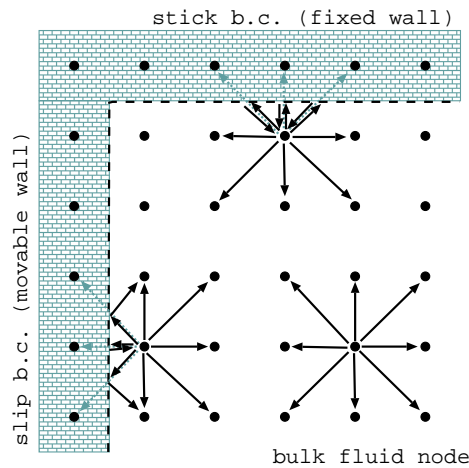


Fig. 3: Schematic implementation of boundary conditions in the LB method.

In the LB method, solid boundary conditions can be implemented by identifying some of the nodes in the lattice as solid nodes. A simple case would be the implementation of a fixed

solid wall, what is implemented by a bounce-back algorithm as illustrated in Fig. 3. During the streaming step, the component of the distribution function that would reach the solid node returns to the fluid node with a velocity in the opposite direction. This rule produces stick boundary conditions at approximately half the distance along the vector joining the fluid and the solid nodes, ensuring that the velocity of the fluid at the wall is precisely the wall velocity. In the cases where the zero velocity plane must be located at a more complicated geometry one should use suited interpolation schemes for each problem [2, 19]. This easy implementation of non-slip boundary conditions supports the idea that LB is quite advantageous for simulating complicated geometries, such as flow through porous media. Nevertheless, it was found such bounce-back condition is only first order accuracy at the boundaries and a several improvements have been proposed to overcome the problem (see [19]).

The second boundary condition illustrated in Fig. 3 corresponds to a solid wall which moves at the same velocity as the fluid in the bulk. In this case, the component of the distribution function that would reach the wall is specularly reflected to the corresponding node.

Initial conditions - As the last aspect I discuss the initial configuration which can be an equilibrium distribution in which all node densities have a constant value. This means that, before any streaming or collision step is performed, all the velocity nodes are vanishing since $\rho(\mathbf{x}, t) = \sum_k f_k^{eq}(\mathbf{x}, t)$. A flow can be induced by imposing constant velocity boundary conditions (see Fig. 3) which approaches a constant flow rate. Another standard initial condition is the assumption of constant pressure. Periodic boundary conditions are particularly useful for modeling bulk systems because they tend to minimize finite size effects.

2.3 Further considerations

The previous implementation give the basics of the hydrodynamic behavior where mass and momentum are conserved quantities, but in order to reproduce the Navier-Stokes equation an extra conservation condition has to be added to Eqs. (10) and (12). This is that the second moment of f_k^{eq} will correspond to the stress tensor Π

$$\Pi = \sum_k f_k^{eq}(\mathbf{x}_i, t) \mathbf{c}_k \mathbf{c}_k, \quad (17)$$

where $\mathbf{c}_k \mathbf{c}_k$ is the diadic product. The stress tensor should have the hydrodynamic form,

$$\Pi = p\mathbb{I} + \rho \mathbf{u} \mathbf{u}, \quad (18)$$

where p is the pressure of the fluid, given by an ideal gas equation of state $p = \rho c_s^2$ in the LB approach, although other equations of state can be implemented. Furthermore, one can show that the described LB procedure yields hydrodynamic behavior in the macroscopic limit, via a Chapman-Enskog expansion [22].

With all these considerations LB constitutes a much more efficient method than its precursor LGA. It has proved to be especially useful in studying flows in complex geometries, like porous media, or the dynamics of colloidal suspensions, as well as in studies of multicomponent systems, and it is extensively employed by a large community.

The model suffers, however, of some intrinsic problems. One is that energy conservation is not fulfilled by the present LB method, such that it is generally restricted to isothermal applications, although there are recent energy conserving generalizations of the model. Another problem

is that 'thermal fluctuations' are not present in the model, while they are in LGA due to the discreteness in the number of particles. This lack of fluctuations can be in some cases an advantage since these are always a source of statistical inaccuracy. But thermal fluctuations are a required element in the correct description of a large number of physical problems, like the Brownian motion of suspensions or in the decay of spontaneous stress fluctuations. For these cases, a possible solution has been proposed based on linear fluctuating hydrodynamics according to which the stress tensor of Eq. (17) should include a noise term. This can therefore be directly added in the simulation code [23].

3 Dissipative particle dynamics (DPD)

Originally, the DPD model was introduced [24] as an attempt to free the Lattice Gas Automata from the lattice. The state of the fluid is described by N particles with continuous positions and velocities. Similar to Molecular Dynamics (MD), the particles time evolution is given by the integration of the Newton's equation of motion.

The first idea of DPD is to consider soft and finite interactions into a standard simulation with MD. The scales of time and space that can be reached are quite large, and phenomena related to processes on mesoscopic scales can be reproduced. The second idea of DPD is that these soft and finite interactions have dissipative and stochastic contributions, as well as a weak conservative term. The introduction of these dissipative and random interactions between DPD particles can be understood if each particle is representing not only one molecule, but rather a group of them. Therefore, a DPD particle models the center of mass of a mesoscopic portion of the fluid, large enough to be a thermodynamic subsystem, but still subjected to thermal fluctuations [25, 26]. The technique has been constructed such that both the number of particles and the total momentum are conserved quantities, as will be outlined in Sec. 3.1. Therefore, there is a transport equation for the momentum density field, coupled to the continuity equation. The macroscopic behavior of this particle model is then hydrodynamic, and not just diffusive as it occurs with Brownian Dynamics.

One of the drawbacks of DPD, as it was originally formulated, is that the total energy of the system is not conserved. It has been shown [25] that such a model cannot sustain temperature gradients, *i.e.* on the same microscopic time scale on which velocities locally relax to equilibrium, the local temperature reaches a spatially uniform equilibrium temperature. The system is thermostatted and can not describe any thermal transport process on macroscopic time scales. The model describes only the essential features of hydrodynamic flows in *isothermal* situations. The DPD model has been generalized [27, 28] by assigning to every DPD particle an internal energy variable such that total energy is globally and locally conserved. Therefore, processes with transport of energy can be considered in addition to momentum transport. This generalization is here referred to as DPD+e and will be introduced in Sec. 3.3.

3.1 Isothermal DPD fluid

In this model the simple fluid is represented by a set of particles, that can be thought of as a lump of fluid, interacting through conservative, dissipative and random forces. The equations

of motion for the positions \mathbf{r}_i and velocities \mathbf{v}_i of the particles are of Langevin type

$$\begin{aligned} d\mathbf{r}_i &= \mathbf{v}_i dt, \\ d\mathbf{v}_i &= \frac{1}{m} \sum_{j \neq i} (\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R) dt, \end{aligned} \quad (19)$$

where i ($i = 1, \dots, N$) labels the particles and m is the mass of a particle. The pair force that particle j exerts on particle i has three contributions: a conservative force \mathbf{F}_{ij}^C , a dissipative force \mathbf{F}_{ij}^D , and a random force \mathbf{F}_{ij}^R . These forces are interpreted as coarse grained averages over microscopic degrees of freedom.

The dissipative and random forces combined act as a thermostat. The dissipative force is proportional to a friction constant and cools the system, whereas the random force heats it up. To qualify as a fluid, DPD should be Galilean invariant and isotropic. The Galilean invariance requires that the forces depend only on relative variables $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ and $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$. Isotropy requires that the forces transform under rotations as vectors. Moreover, the drift term of the Fokker-Planck equation must be linear in the velocity variable, and the diffusion term independent of it [29]. These requirements are satisfied if the dissipative force F^D is linear in the velocities and the random force F^R is independent of the velocity. A simple form of the forces satisfying these criteria is

$$\begin{aligned} \frac{1}{m} \mathbf{F}_{ij}^D dt &= -\gamma w_D(r_{ij}) (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij}) \hat{\mathbf{r}}_{ij} dt, \\ \frac{1}{m} \mathbf{F}_{ij}^R dt &= \sigma w_R(r_{ij}) \hat{\mathbf{r}}_{ij} dW_{ij}, \end{aligned} \quad (20)$$

where r_{ij} and $\hat{\mathbf{r}}_{ij}$ are respectively the modulus and the unit vector parallel to \mathbf{r}_{ij} . The coefficients γ and σ are positive constants that control the friction and noise amplitudes. The range interaction functions w are bounded positive functions of the relative distance r_{ij} , and vanish for $r > r_c$.

The physical interpretation of the dissipative force \mathbf{F}_{ij}^D is as follows. When particles i and j are approaching/receding, the quantity $(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})$ is negative/positive which implies that both particles feel a viscous force slowing down their relative motion in the $\hat{\mathbf{r}}_{ij}$ direction. Physically, the friction is related to the viscous interaction of different parts of the fluid.

The random force is a physical consequence of the mesoscopic description, and it is postulated through independent increments of the Wiener process. These processes describe Gaussian white noise such that $\langle dW_{ij}(t) \rangle = 0$. The symmetry property $\mathbf{F}_{ij} = -\mathbf{F}_{ji}$ ensures that the total momentum is conserved, and enforces that $dW_{ij} = dW_{ji}$. The Wiener processes are interpreted through the Ito calculus rule, and normalized as

$$dW_{ij} dW_{i'j'} = (\delta_{ii'} \delta_{jj'} + \delta_{ij'} \delta_{ji'}) dt, \quad (21)$$

i.e. $dW_{ij}(t)$ is an infinitesimal of $\mathcal{O}(dt^{1/2})$ [30]. This form respects the symmetry under particle interchange. From a practical point of view, this term is calculated as $dW_{ij} = \zeta_{ij} \sqrt{dt}$, where $\zeta_{ij} = \zeta_{ji}$ is a random number which can be uniformly generated in the interval $(-1, 1)$. It can also be a Gaussian distributed number in the same interval, but it has been checked that it is not necessary.

Furthermore, the relation between the dissipative and the random forces has a precise form in order that the system in equilibrium displays the Maxwell-Boltzmann distribution. This implies the fluctuation dissipation relations [25, 31]

$$m\sigma^2 = 2\gamma k_B T, \quad w_D(r) = w_R^2(r) \equiv w(r), \quad (22)$$

where a single range function has been defined.

The possible conservative forces on these mesoscopic length scales are supposed to be non-singular. Theoretical studies [25, 31] impose no further restrictions on the functional form of the conservative forces. It should be noted that many advantages of the model come from the soft interactions since they allow large time steps. In this spirit the conservative force is taken in the majority of the cases as

$$\frac{1}{m} \mathbf{F}_{ij}^C dt = -\frac{1}{m} \frac{\partial \phi(r_{ij})}{\partial \mathbf{r}_{ij}} dt = a w_C(r_{ij}) \hat{\mathbf{r}}_{ij} dt \quad (23)$$

where $\phi(r_{ij})$ is the pair potential, the coefficient a controls the strength of the conservative repulsion, and $w_C(r_{ij})$ is a bounded function similar to $w_D(r)$. The force is then weak, repulsive and of finite range. Note that the conservative forces have been re-cast in the form of real pressure forces [32–34].

Detailed balance and Fokker-Planck equation - The Langevin equations (19)-(21) constitute a mathematically well-defined set of stochastic differential equations [30]. Using standard procedures (see Chapter B4) the corresponding Fokker-Planck equation, governing the time evolution of the N particle distribution function, can be derived [31],

$$\partial_t \rho(\mathbf{X}, t) = L \rho(\mathbf{X}, t) = \left[-\sum_i \mathbf{v}_i \cdot \nabla_i + \sum_{i < j} T(ij) \right] \rho(\mathbf{X}, t), \quad (24)$$

where $\mathbf{X} = \{\mathbf{x}_i = (\mathbf{r}_i, \mathbf{v}_i) | i = 1, \dots, N\}$ is the phase-space vector, and $\nabla_i = \partial/\partial \mathbf{r}_i$. The Fokker-Planck operator $T(ij)$ is the sum of a conservative and a dissipative contribution

$$T(ij) = T^C(ij) + T^D(ij), \quad (25)$$

$$T^C(ij) = -\frac{\mathbf{F}_{ij}^C}{m} \cdot \partial_{ij} = \frac{1}{m} \frac{\partial \phi(r_{ij})}{\partial \mathbf{r}_{ij}} \cdot \partial_{ij},$$

$$T^D(ij) = \gamma w(r_{ij}) \hat{\mathbf{r}}_{ij} \hat{\mathbf{r}}_{ij} : \left[\partial_{ij} \mathbf{v}_{ij} + \frac{k_B T}{m} \partial_{ij} \partial_{ij} \right]. \quad (26)$$

where velocity derivatives are denoted as $\partial_i = \partial/\partial \mathbf{v}_i$, and $\partial_{ij} = \partial_i - \partial_j$. Here, the detailed balance condition of Eq.(22) has been imposed, *i.e.*, the stationary equilibrium solution $\partial_t \rho_0(\mathbf{X}) = 0$ of Eq.(24) should be the Gibbs state (thermal equilibrium)

$$\rho_0(\mathbf{X}) = \frac{1}{Z_N} \exp \left[-\frac{1}{k_B T} \left(\sum_i \frac{m}{2} v_i^2 + \sum_{i < j} \phi(r_{ij}) \right) \right]. \quad (27)$$

where Z_N is the normalizing partition function.

Interaction range function - In principle, conservative and dissipative processes are not related such that the interaction range functions can be independently chosen. However, for practical reasons, they are usually considered to be the same, $w_C(r) = w_D(r) \equiv w(r)$, or $w_C(r) = w_R(r)$. On the other hand, several choices have been performed for the function $w(r)$, most of which are illustrated in Fig. 4. Furthermore, two different normalizations of the range function have been used in the literature. Some authors choose the value at the origin

of the range function $w(r)$ to be a number of $\mathcal{O}(1)$ while others normalize it with the number density ρ by choosing $\rho \int dr \bar{w}(r) \equiv \rho[\bar{w}] = 1$. Here, these choices are respectively denoted by $\{\gamma, w(r)\}$ and $\{\bar{\gamma}, \bar{w}(r)\}$. It is known that the friction functions, $\gamma w(r)$ and $\bar{\gamma} \bar{w}(r)$, must be the same. Therefore, the same physical friction is described by different numerical values of the friction *coefficient* in different normalizations, or choices of the interaction range function, and all of them can be compared by, $\rho[w]\gamma = \rho[\bar{w}]\bar{\gamma} = \bar{\gamma}$.

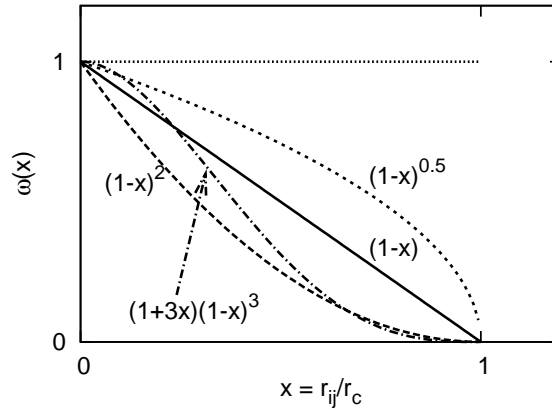


Fig. 4: Common choices for the interaction range function in DPD

Integration algorithms - The DPD model has been defined as continuous in positions and momenta, but discrete in time, since discrete time steps are required to update the variables in Eq.(19). In principle, all update methods known from MD could be used, but in practice the pairwise coupling of particles through dissipative and random forces makes the integration of the equations a non-trivial task. The first method employed was the Euler scheme, but the equilibrium temperature, defined through Eq.(22), was recovered only in the limit of zero time step [35]. The main difficulties come from the dissipative forces that explicitly depends on the relative particle velocities which depend themselves on the dissipative and random forces. Quite an improvement came from modifications of the velocity Verlet-type algorithm [36,37], or from a self consistent algorithm [38], based on the leap-frog scheme where time reversibility is preserved. As an alternative, other works [39,40] suggest time step dependent model parameters in order to control the equilibrium behavior.

The most recent and accurate developments in DPD integrators [41–43] come from applying schemes commonly used in solving stochastic differential equations. The main idea [41] is to factorize the integration process such that terms coming from the conservative forces are calculated separately from the dissipative and random terms. The conservative part is solved then by standard MD methods, while the fluctuation-dissipation terms are solved with methods coming from Trotter or Strang expansions which allow one for deriving symplectic algorithms like those in Chapter B3. For details of the implementation for DPD see the original papers [41–43].

DPD units and parameters - In the previous description of the DPD model no preferential units reference frame has been chosen. The most common set of units is

Mass unit:	m	\equiv particle mass
Length unit:	r_c	\equiv interaction range
Energy units:	$k_B T$	\equiv system temperature

such that time would be $r_c \sqrt{k_B T / m}$ in this units.

Other parameters that the method requires to be specified are the particle density ρ , the friction coefficient γ , the repulsion parameter a , and the simulation box size L . The particle density ρ is the number of particles in a volume unit, r_c^3 , such that the number of interacting pairs for each particle will be $(4\pi/3)\rho r_c^3$, and the CPU time required per time step will increase quadratically with ρ . It is therefore convenient to choose not very large values and $\rho = 3, 4$ are most common ones. The value of the friction coefficient γ and therefore the noise amplitude $\sigma = \sqrt{2\gamma k_B T / m}$ determine the mechanical properties of the fluid, but they are also inversely proportional to the time that the system takes to relax to equilibrium. It would be then convenient to take high values of γ although too high values have shown to give stability problem. Standard values are $\gamma = 4.5, 7$. The previous parameter choices of ρ and γ are mainly due to a quite detailed study [36] where the stability and efficiency of the velocity Verlet-type DPD algorithm was explored, such that these conclusions could slightly vary with the more recent symplectic algorithms mention previously.

Transport properties - Explicit expressions for the transport properties of the DPD fluid exists only in the case where the conservative forces are not present, $F_{ij}^C = 0$ which corresponds to an ideal gas [25, 44]. They can be expressed in terms of the thermal velocity $v_0 = \sqrt{k_B T / m}$ and the characteristic collision frequency $\omega_0 = \rho[w]\gamma/d$, as

$$\begin{aligned} \text{Diffusion coefficient: } D &= \frac{v_0^2}{\omega_0} \\ \text{Kinematic viscosity: } \nu &= \frac{v_0^2}{2\omega_0} + a_2 \omega_0 r_c^2 \end{aligned} \tag{28}$$

where a_2 is a geometrical prefactor $a_2 = [r^2 w] / (2(d+2)r_c^2[w])$, with $[r^2 w] \equiv \int d\mathbf{r} r^2 w(r)$. Although these expressions are limited in their applicability, they come to indicate that the momentum transport is of the same order of magnitude as the mass transport. This is quantified by the dimensionless Schmidt number $Sc = \nu/D$, which from the above expression and the common parameter values in DPD is of the order of unity while in real liquids as water this number is of the order of 10^3 . Higher values of Sc could be obtained by increasing the coefficient γ although the system stability would have to be revised then.

The repulsion parameter - The last model parameter to be discussed is the strength of the conservative force a . Apart from the trivial value for the ideal gas, the most extended choice is the one that matches the value of the dimensionless isothermal compressibility χ , with the value of the same quantity which is known for water. Measurements of the pressure obtained with a DPD fluid allow to determine χ through the density dependence of the pressure as a function of the parameter a . In this way [36], $a = 75k_B T / \rho$ is determined as the parameter to reproduce water. In a further step, different species can be introduced into a DPD model. In these cases, the repulsion parameter is supposed to depend on the interacting parts a_{ij} , such that extra repulsive or attractive interactions between unlike componets can easily be considered.

3.2 Applications

In order to model complex fluids with DPD, the general strategy has been to introduce additional interactions between dissipative particles in order to define mesostructures. For example, sets of particles are joined together with harmonic forces to obtain a coarse grained model for linear polymer molecules. Further, a solvent can be represented by other (not connected) dissipative particles in order to model a dilute polymer solution. Colloidal particles of arbitrary shape can be modeled by “freezing” dissipative particles within a region that moves then as a rigid body. These solid objects coexist with solvent particles and strongly modify the rheology of the system. Introducing two types of particles that interact differently allows to model mixtures and study, for example, spinodal decomposition. Given the simplicity of modeling mesostructures, DPD appears as a competitive technique in the field of complex fluids. In this section, a brief review is given of some applications of DPD which can be found in the literature.

Rheology of colloidal suspensions - This was the first application of the method, already presented in the same paper where DPD was introduced [24, 45]. The solvent is simulated by bare DPD particles and additional constraints are imposed to obtain large solid objects like suspended spheres. They can be viewed as a locally “frozen” portion of the fluid, where the particles forming the solid object remain at fixed relative positions, while the object as a whole moves, as a consequence of the forces exerted by the fluid particles surrounding the object. It is interesting to note that the proper no-slip boundary condition for the fluid motion emerges at the solid boundaries, without additional modeling, thanks to the dissipative force.

Measurements of the viscosity are performed for different volume fractions and compared with experimental results. The agreement was quite good for volume fractions below 30%, which is already a regime where other simulation methods fail. However, later studies [46] explore regimes of higher volume fractions where different problems appear: Stick boundary conditions between the colloid and the solvent are not fully satisfied anymore. Colloid particles are defined as soft balls which can interpenetrate and, therefore, the radius of the colloidal particle is not well defined. At high volume fractions solvent particles are expelled from the region in between two colloidal particles. Therefore, the interaction is no longer hydrodynamically mediated and depletion forces appear. Routes for characterizing and solving these problems have been worked out by different groups [47, 48].

Rheology of dilute polymers - Polymer molecules are constructed in DPD by linking several dissipative particles through springs [49]. Dilute polymer solutions are modeled by a set of polymer molecules interacting with a sea of fluid particles. The solvent quality can be varied by fine tuning the solvent-solvent and solvent-monomer interactions. In this way, a collapse transition has been observed in passing from a good solvent to a poor solvent [50]. DPD polymers can freely pass through each other as ‘phantom chains’. For melts and dense solutions of polymers, scaling laws for the radius of gyration and relaxation time have been studied in Ref. [51]. The effects of screening of hydrodynamic and excluded volume interactions in the melt can be observed, in satisfactory agreement with the Kirkwood theory. The model is unable to simulate entanglements due to the soft interactions between beads that allow polymer crossing, although this effect can be partially controlled by suitable adjusting the length and intensity of the springs [37]. Hydrodynamic interactions and excluded volume interactions are then displayed depending on the quality of the solvent. Rheological properties have also been studied showing a good agreement with known kinetic theory results [50, 52].

Phase separation - Fluid mixtures are modeled in DPD by assuming two types of particles. In order to reproduce an immiscible mixture unequal particles repel each other more strongly than equal particles thus favoring phase separation. Starting from random initial conditions representing a high temperature miscible phase suddenly quenched, the domain growth has been investigated. In 2D symmetric mixtures (equal fraction of each fluid), a crossover from a domain growth scaling as $t^{1/2}$ to $t^{2/3}$ is observed. This signals a crossover from a diffuse to an inertial hydrodynamic regime. For asymmetric quenches the growth law scales as $t^{1/2}$ [53].

It should be noted that in Ref. [54] a 3D symmetric mixture showed a lack of scaling which corresponds to a viscous dominated growth. This was later on elucidated through lattice Boltzmann simulations to be due, most probably, to finite size effects [55]. Although lattice Boltzmann simulations allow to explore larger time scales than DPD for this system [55], the simplicity of DPD modeling allows to generalize easily to more complex systems in a way that can not be done through the lattice Boltzmann method. For example, mixtures of homopolymer melts have been modeled with DPD [36]. Surface tension measurements allow for a mapping of the model to the Flory-Huggins theory [36]. In this way, thermodynamic information has been used to fix the model parameters of DPD. A more detailed analysis of this procedure has been presented in Refs. [56, 57] where a calculation of the phase diagram of monomer and polymer mixtures of DPD particles allows one to study the connection between the difference in the repulsion parameters between monomers and polymers on the one hand, and the Flory-Huggins parameter χ on the other hand.

Another successful application of DPD has been the simulation of microphase separation of diblock copolymers [58], which allows one to investigate the pathway to equilibrium. The hydrodynamic interactions [59] were shown to be a critical part in the kinetics of the mesophase formation. Qualitative agreement with experiments and with existing mean-field theory for symmetric and asymmetric polymers is obtained, and clues about the quantitative differences can be found.

Other complex fluid systems - DPD has been applied to more complex situations, like the dynamics of a drop at a liquid-solid interface [60], flow and rheology in the presence of polymers grafted to walls [61], colloidal adsorption onto polymer coated surfaces [37], amphiphilic mesophases [62], model membranes [63] or geometrical packing of filler in composites [64].

3.3 Energy conserving DPD model (DPD+e)

The energy conserving DPD model [27, 28] is a generalization of the preceding DPD where the state of each particle is described by three variables, position \mathbf{r} , velocity \mathbf{v} and a new variable ϵ that accounts for the internal energy. In the picture where dissipative particles are understood as droplets or mesoscopic clusters of microscopic particles, one can consider the kinetic energy lost in dissipative interactions as being transformed into energy of internal degrees of freedom of a particle. The number of internal states with energy ϵ , $\exp[s(\epsilon)/k_B]$, is modeled by an entropy function $s(\epsilon)$, implying a temperature $T(\epsilon)$ defined through $\partial s(\epsilon)/\partial \epsilon = 1/T(\epsilon)$. One then constructs an equation of motion for the internal energy state variable ϵ_i , such that the total energy E of the N particle system is conserved,

$$E = \sum_i \left(\epsilon_i + \frac{1}{2} m v_i^2 \right) + \sum_{i < j} \phi(r_{ij}). \quad (29)$$

The idea is then to impose that all dissipated mechanic energy is stored in internal degrees of freedom of the particles, in such a way that the total energy $E_{tot} = E_{mec} + \sum \epsilon_i$ is exactly conserved. Let $E_{mec} = \sum_{i \neq j} \frac{1}{2} \phi(r_{ij}) + \sum_i \frac{1}{2} m v_i^2$ be the mechanical energy. Then an 'infinitesimal' increment of the total energy can be evaluated as

$$\begin{aligned} dE_{mec} &= E_{mec}(\mathbf{r} + d\mathbf{r}, \mathbf{v} + d\mathbf{v}) - E_{mec}(\mathbf{r}, \mathbf{v}) \\ &= - \sum_{i \neq j} \mathbf{F}_{ij}^C d\mathbf{r}_{ij} + \sum_i m \mathbf{v}_i d\mathbf{v}_i + m \sum_i d\mathbf{v}_i d\mathbf{v}_i \equiv (d\epsilon_i)^{VH}. \end{aligned} \quad (30)$$

Note that the 'second' order term is a consequence of stochastic calculus, where $d\mathbf{v}_i$ contains terms of $\mathcal{O}((dt)^{1/2})$ (see Eq.(21)). Substitution of the equation of motion (19) and use of the Ito rule (21), yield for Eq.(30) to the loss of energy due to *viscous heating* (VH) is

$$(d\epsilon_i)^{VH} = \frac{m}{2} \sum_j [-\gamma_{ij} w_D(r_{ij}) (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})^2 + \sigma_{ij}^2 w_R^2(r_{ij})] dt + \frac{m}{2} \sum_j \sigma_{ij} w_R(r_{ij}) (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij}) dW_{ij} \quad (31)$$

Note that from here on the sums over j satisfy the constraint $j \neq i$. This expression describes how the friction forces contribute to the change of mechanical energy. In addition the phenomenon of *heat conduction* (HC) has to be considered, where temperature differences between particles (subsystems) produce a flux of internal energy. To formulate stochastic differential equations for the phenomenon of heat conduction, a discrete fluctuating Fourier equation of heat conduction (HC) is considered [65, 66],

$$(d\epsilon_i)^{HC} = \sum_j \kappa_{ij} w_D^\epsilon(r_{ij}) (T_j - T_i) dt + \sum_j \alpha_{ij} w_R^\epsilon(r_{ij}) dW_{ij}^\epsilon. \quad (32)$$

The first term on the r.h.s. is deterministic and specifies that a temperature difference causes flow of energy. The second term is stochastic and takes into account thermally induced fluctuations in each particle due to random interchange of energy between particles [10]. The factor κ_{ij} modulates the intensity of thermal conduction and α_{ij} is the amplitude of the noise. These functions are assumed to be symmetric under particle interchange. Both terms in Eq.(32) must be antisymmetric under particle interchange for the total energy of the system to remain conserved. Consequently, the increments of the Wiener process associated with the heat conduction have then to be antisymmetric under particle interchange $dW_{ij}^\epsilon = -dW_{ji}^\epsilon$. The Ito rule is expressed as

$$dW_{ij}^\epsilon dW_{i'j'}^\epsilon = (\delta_{ii'} \delta_{jj'} - \delta_{ij'} \delta_{ji'}) dt. \quad (33)$$

The two Wiener processes dW_{ij} in VH, and dW_{ij}^ϵ in HC are uncorrelated.

The equation of motion for the internal energy follows by summing the terms in Eqs.(31) and (32), i.e. $d\epsilon_i = (d\epsilon_i)^{VH} + (d\epsilon_i)^{HC}$. To summarize, the full set of stochastic differential equations for $\mathbf{x}_i = \{\mathbf{r}_i, \mathbf{v}_i, \epsilon_i\}$ is the same as in Eq.(19) supplemented with the equation of motion for ϵ_i . The equilibrium distribution function is a generalization of the Gibbs state of Eq.(27) given by,

$$\rho_0(\mathbf{X}) = \frac{1}{Z_N} \exp \left[-\frac{1}{k_B T} \sum_i \left(\frac{m}{2} v_i^2 + \epsilon_i + \frac{1}{2} \phi(r_{ij}) \right) + \frac{1}{k_B} \sum_i s(\epsilon_i) \right]. \quad (34)$$

where the \mathbf{X} is a point in the phase space, $\mathbf{X} = \{\mathbf{x}_i = (\mathbf{r}_i, \mathbf{v}_i, \epsilon_i) | i = 1, \dots, N\}$, Z is the normalization factor. In order to ensure that the system relaxes to the previous equilibrium

distribution, fluctuation dissipation relations, similar to Eqs.(22), are now necessary

$$\begin{aligned} 2k_B\gamma_{ij} &= \frac{1}{2}m\sigma_{ij}^2 \left(\frac{1}{T_i} + \frac{1}{T_j} \right) \quad , \quad w_D(r) = w_R^2(r) \equiv w(r) \\ 2k_B\kappa_{ij}T_iT_j &= \alpha_{ij}^2 \quad , \quad w_D^\epsilon(r) = (w_R^\epsilon)^2(r) \equiv w(r). \end{aligned} \quad (35)$$

Furthermore, the strength coefficients can respectively have the form $\kappa_{ij} = \alpha\kappa_0k_B$, and $\sigma_{ij} = \sigma_0$, where σ_0 and κ_0 are positive constants having the dimension of a frequency. The range functions $w(r)$ for C, VH, and HC could in principle be different, but for simplicity they are generally chosen to be the same. Further details of this model can be found in Refs. [67,68]

Equation of state - To define the DPD+e model completely, one has to specify the entropy function $s(\epsilon)$ as a function of the internal energy of a DPD particle, *i.e.* its density of internal states. The simplest choice is the entropy for an ideal solid

$$s(\epsilon) = C_v \ln(\epsilon/\epsilon_u), \quad (36)$$

where $C_v = \alpha k_B$ is the heat capacity of a DPD particle, which is assumed to be a constant independent of ϵ . The parameter ϵ_u is a constant with dimension of energy. This constant is necessary in order to have a dimensionless argument for the logarithm, but it is irrelevant as it represents an additive constant in the entropy. By differentiating Eq.(36), the relation between the internal temperature of a particle T_i and its internal energy ϵ_i are obtained,

$$\epsilon_i = C_v T(\epsilon_i) = \alpha k_B T_i. \quad (37)$$

The dimensionless number $\alpha = C_v/k_B$ is a measure of the “size” of the DPD particle because it scales like the number of internal degrees of freedom of the particle. Hence, α is a large number.

Later extensions of the original DPD model have been proposed in order to include various aspects. One has been the so called Fluid Particle Model [69]. This model has a more general dissipative force which is not central. For this reason it is necessary to include a spin variable. This new variable ensures that the total angular momentum of the system is conserved. Another generalization has been introduced for modeling viscoelastic flows [70] by including an elastic variable for each fluid.

4 Multiparticle collision dynamics (MPC)

This mesoscale simulation technique is a variant of the DSMC method, in which binary collisions are replaced by multi-particle collisions in a prescribed collision volume. This method has been called multi-particle-collision dynamics (MPC) or stochastic rotation dynamics (SRD). It employs a discrete-time dynamics with continuous velocities and local multi-particle collisions. Mass, momentum, and energy are locally conserved quantities by construction and it has been demonstrated that the hydrodynamic equations are satisfied. MPC is also the most recent technique of the ones treated in this chapter, since it was first introduced by Malevanets and Kapral [71,72] in 1999.

4.1 MPC fluid model

The fluid is modeled by N point particles. Each of these particles is characterized by its position \mathbf{r}_i and velocity \mathbf{v}_i , and labeled with $i = 1, \dots, N$. Positions and velocities are continuous variables, which evolve in discrete increments of time. The mass m_i associated with the particles is usually taken to be the same, but more generally, different masses can be assigned.

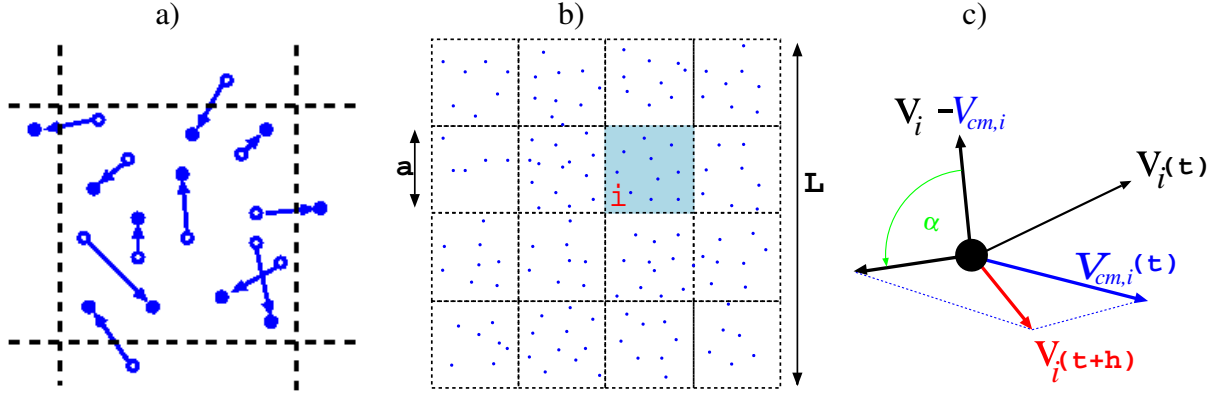


Fig. 5: Diagram of the MPC dynamics in 2 dimensions. (a) Streaming step, (b) particles sorted into collision boxes and (c) rotation of the particle velocity relative to the center of mass.

The MPC algorithm consists of two steps, streaming and collision, which are illustrated in Fig. 5. In the streaming step the particles do not interact with each other (see Fig. 5a), they move ballistically according to their velocities during a time increment h , to which I will refer as *collision time*. Thereby, the evolution rule is

$$\mathbf{r}_i(t+h) = \mathbf{r}_i(t) + h\mathbf{v}_i(t). \quad (38)$$

In the collision step, the particles are sorted into collision boxes (see Fig. 5b), and interact with *all* other particles in the same collision box. This multibody interaction takes place through the collision box center of mass velocity. This is

$$\mathbf{v}_{cm,i}(t) = \frac{\sum_j^{(i,t)} (m_j \mathbf{v}_j)}{\sum_j m_j}, \quad (39)$$

such that $\mathbf{v}_{cm,i}(t)$ is the velocity of the center of mass of all particles j , which are located in the collision box of particle i at the considered time t . The collision boxes are typically the unit cells of a d -dimensional cubic lattice with lattice constant a , although other geometries would be possible. The collision is then defined as a rotation of the velocities of all particles in a box in a co-moving frame with its center of mass. Thus, the velocity of the i -th particle after the collision is

$$\mathbf{v}_i(t+h) = \mathbf{v}_{cm,i}(t) + \mathcal{R}(\alpha) [\mathbf{v}_i(t) - \mathbf{v}_{cm,i}(t)], \quad (40)$$

where $\mathcal{R}(\alpha)$ is a stochastic rotation matrix. This implies that each particle changes during the collision the magnitude and the direction of its velocity (see Fig. 5c), in such a way that the total momentum and kinetic energy are still conserved within every collision box. This is easy to visualize since the collision box center of mass velocity $\mathbf{v}_{cm,i}$ does not change during the

collision,

$$\begin{aligned} \sum_j^{(i,t)} m \mathbf{v}_j(t+h) &= \sum_j^{(i,t)} m (\mathbf{v}_{cm,i}(t) + \mathcal{R}(\alpha) [\mathbf{v}_j(t) - \mathbf{v}_{cm,i}(t)]) = \sum_j^{(i,t)} m \mathbf{v}_j(t) \\ \sum_j^{(i,t)} \frac{m}{2} v_j^2(t+h) &= \sum_j^{(i,t)} \frac{m}{2} (\mathbf{v}_{cm,i}(t) + \mathcal{R}(\alpha) [\mathbf{v}_j(t) - \mathbf{v}_{cm,i}(t)])^2 = \sum_j^{(i,t)} \frac{m}{2} v_j^2(t). \end{aligned} \quad (41)$$

Therefore, with the collision rule in Eq. (40), the conservation of mass, local momentum and kinetic energy are guaranteed by construction.

MPC units - In the simulations, N particles are initially placed at random in a cubic system of linear extension L . The average number of particles in a collision box is $\rho = N(a/L)^d$, the scaled number density. Starting from an arbitrary distribution of velocities, only a few steps are required to reach the Maxwell-Boltzmann velocity distribution. The equilibrium temperature T is then given by the average kinetic energy $m \langle \mathbf{v}_i^2 \rangle = 3k_B T$, where k_B is the Boltzmann constant. It is necessary to choose reference units, and these can for instance be,

$$\begin{aligned} \text{Mass unit:} \quad m &\equiv \text{particle mass} \\ \text{Length unit:} \quad a &\equiv \text{collision box size} \\ \text{Energy units:} \quad k_B T &\equiv \text{system temperature} \end{aligned}$$

which corresponds to measure length and time according to $\hat{x} = x/a$ and $\hat{t} = t\sqrt{k_B T/ma^2}$. The scaled mean free path is then given by $\lambda = \hat{h}$.

Random rotation - Together with the collision step the stochastic rotation matrix $\mathcal{R}(\alpha)$ has been introduced, such that α is a parameter of the model. In two dimensions, the rotation of the relative velocity is simply given by an angle $\pm\alpha$, where the sign independent and randomly chosen for each cell. In three dimensions, I describe two main schemes for the random collisions. The first one [73] chooses the rotation direction among the three main axis and the rotation is performed by an angle $\pm\alpha$. The second scheme [74] consists in choosing a random direction in space for each box around which the relative velocities are rotated by the angle α . The second scheme is slightly more difficult to implement but in return it introduces less anisotropy in the system due to the underling lattice.

Rotation of a 3D vector around a random direction A random direction is independently generated in each collision box by selecting two uncorrelated random numbers r_1, r_2 from a distribution in the interval $[0, 1]$. The random unity vector \mathbf{R} has components,

$$R_x = \sqrt{1 - \varphi_1^2} \cos \varphi_2, \quad R_y = \sqrt{1 - \varphi_1^2} \sin \varphi_2, \quad R_z = \varphi_1 \quad (42)$$

where $\varphi_1 = 2r_1 - 1$ and $\varphi_2 = 2\pi r_2$. The rotation of \mathbf{v} around \mathbf{R} by an angle α is denoted as $\xi \equiv \mathcal{R}(\alpha)\mathbf{v}$ and sketched in Fig. 6a. The illustrated rotation transforms the vector $\mathbf{v} = \mathbf{v}_{\parallel} + \mathbf{v}_{\perp}$ onto $\xi = \mathbf{v}_{\parallel} + \xi_{\perp}$, where $\mathbf{v}_{\parallel} = (\mathbf{R} \cdot \mathbf{v})\mathbf{R}$, and $\mathbf{v}_{\perp} = \mathbf{v} - \mathbf{v}_{\parallel}$. The rotated vector can be calculated as,

$$\xi = \mathbf{v}_{\parallel} + \mathbf{v}_{\perp} \cos \alpha + (\mathbf{v}_{\perp} \times \mathbf{R}) \sin \alpha. \quad (43)$$

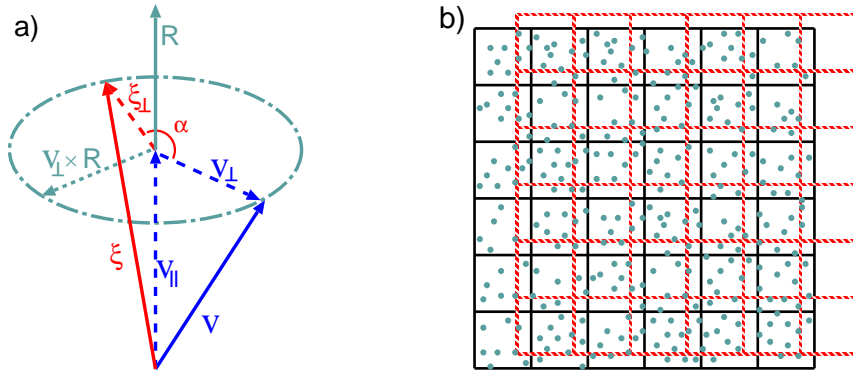


Fig. 6: (a) Rotation of a vector \mathbf{v} around a direction given by a unit vector \mathbf{R} . (b) Diagram of the random shift of the collision grid.

Random shift - In order to perform the multiparticle collision, as I have already discussed, particles are sorted into cells where the collision take place. The choice of these collision boxes defines a preferential grid, and the first naive choice would be a fixed grid whose outside borders coincide with the system boundaries, similar as sketched in Fig. 5b. Nevertheless, such a fixed grid does not fulfill Galilean invariance if the mean free path is not large enough. This can be visualized in the case of two systems, one at rest and the second moving with a constant velocity. If the displacement of particle i is smaller than the size of the collision box a , the particles in the collision box will not be the same in the moving system and in the system at rest, which will lead to different dynamics in the two cases and therefore a breakdown of the Galilean invariance. Therefore, a *random shift* of the collision grid has to be performed in the execution of the collision step [75, 76], in order to ensure Galilean invariance in the full range of possible parameters.

Random shift is performed by displacing the collision grid a random number uniformly distributed in the interval $(0, 1)$ which is chosen independently in each collision. In Fig.6b, the solid grid represents the fixed grid, while the discontinuous grid would be one of the possible displaced grids. Note that periodic boundary conditions (see Chapter B3) are applied, such that for example, the last-right column of the shifted grid would include some particles of the first-left column of the fixed grid. As a consequence of such a shift no special frame exists and Galilean invariance is restored. Similarly, two particles placed at a quite small distance would not interact if a fix grid would separate them. The random shift implementation produces also that the probability of two particles to interact will be inversely proportional to their relative distance, in a way similar to a soft range potential. Random shift also facilitates the transfer of momentum between neighboring particles, since the different positions of the grid in two consecutive collision steps make possible transfer of momentum over larger distances per time unit.

Implementation of walls Many interesting require the implementation of walls as boundary conditions. A simple case would be the implementation of a fixed solid wall. This is performed by applying the so called stick boundary conditions, as have already been introduced for the LB method. For simulating fixed walls with MPC, standard bounce-back is applied during the streaming step, such that when a particle hits the walls it returns in the incoming direction with inverted velocity (see Fig. 7a), and this will be enough when the walls exactly coincide with the

boundaries of the collision cells. However, the walls will generally not coincide with the cell boundaries, due to special geometries or to random shift. In these cases (see Fig. 7b), the cells in the boundary will be generally partially filled, which will not lead to the desired stick boundary conditions. An efficient solution to overcome this problem has been proposed [77]. The idea is that for all the cells of the channel which are cut by walls and have as a consequence a number of particles n smaller than the average number of particles in the bulk ρ , extra *virtual particles* will be added. The function of these virtual particles is to obtain the behavior of a fluid with constant density and temperature in the cells at the boundaries. The velocities of the virtual-wall particles are drawn from a Maxwell-Boltzmann distribution of zero average velocity and the same temperature T as the fluid. The collision step (40) is then carried out with the average velocity of all particles in the cell. Since the sum of random vectors drawn from a Gaussian distribution is again Gaussian-distributed, it is not necessary to calculate the velocities of the individual particles. Instead, the center of mass velocity in Eq.(40) can be written as

$$\mathbf{v}_{cm,i} = \frac{\sum_j^i m \mathbf{v}_j + \mathbf{a}}{\rho} \quad (44)$$

where \mathbf{a} is a vector whose components are numbers from a Maxwell-Boltzmann distribution with zero average and variance $(\rho - n)k_B T$.

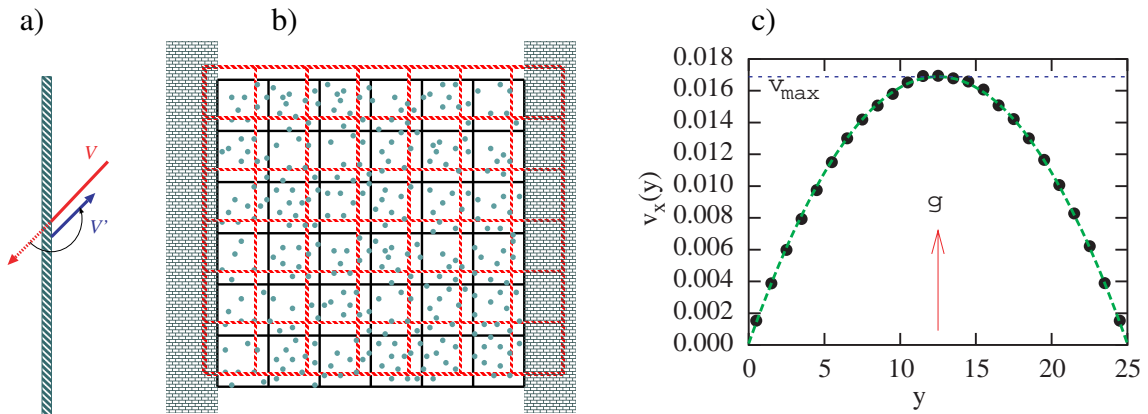


Fig. 7: a) Diagram of the bounce-back rule. b) Random-shift in the presence of walls. c) Parabolic velocity profile in the presence of a gravitational field reproduced by MPC.

Poiseuille flow Giving a fluid resting between two planar walls, a gravitational field g is applied in one direction parallel to the walls. After a relaxation time, the system reaches a stationary state with a parabolic velocity profile between the walls and in the direction of the force. See an example in Fig. 7c, where planar fixed walls are implemented at $y = 0$ and $y = L_y = 25$. It is known [78] that the measured maximum velocity of the parabola is inversely proportional to the kinematic viscosity ν of the fluid like $v_{max} = gL_y^2/(8\nu)$. This behavior is reproduced by MPC simulations, and it can be used as a measurement for the viscosity displayed by the MPC fluid [77, 79, 80]. Alternative methods to determine the viscosity from simulations have been employed in Refs. [81] and [82], where a system under shear flow and vorticity correlations have been respectively used.

4.2 Transport properties

Transport coefficients of the MPC solvent have been studied intensively. Analytical expressions have been derived from kinetic theory by generalizing point-like collisions to finite collision volumes [73, 76, 81, 82]. The theoretical expressions describe numerical results very well. This is quite convenient since different properties of the fluid can be tuned by choosing the adequate set of model parameters.

Viscosity - The total kinematic viscosity, $\nu = \nu_{kin} + \nu_{coll}$, is the sum of two contributions, the kinetic viscosity ν_{kin} and the collisional viscosity ν_{coll} , which have been calculated in two and three dimensions. In three dimensions [73, 81], these expressions are

$$\begin{aligned}\nu_{coll} &= \frac{1}{\lambda} \frac{(1 - \cos \alpha)}{18} \left(1 - \frac{1}{\rho}\right) \\ \nu_{kin} &= \lambda \left[\frac{1}{(4 - 2 \cos \alpha - 2 \cos 2\alpha)} \frac{5\rho}{\rho - 1} - \frac{1}{2} \right].\end{aligned}\quad (45)$$

The viscosity data obtained from Poiseuille flow simulations are presented in Fig. 8 together with the theoretical predictions of Eq. (45). The obtained agreement is quite remarkable, in contrast to the case of other mesoscopic simulation techniques such as dissipative particle dynamics [38]. Density fluctuations can also be included in the theory [81], which noticeably improves the agreement with the simulations results for small number densities; for $\rho = 5$ and $\rho = 10$, these contributions are negligible.

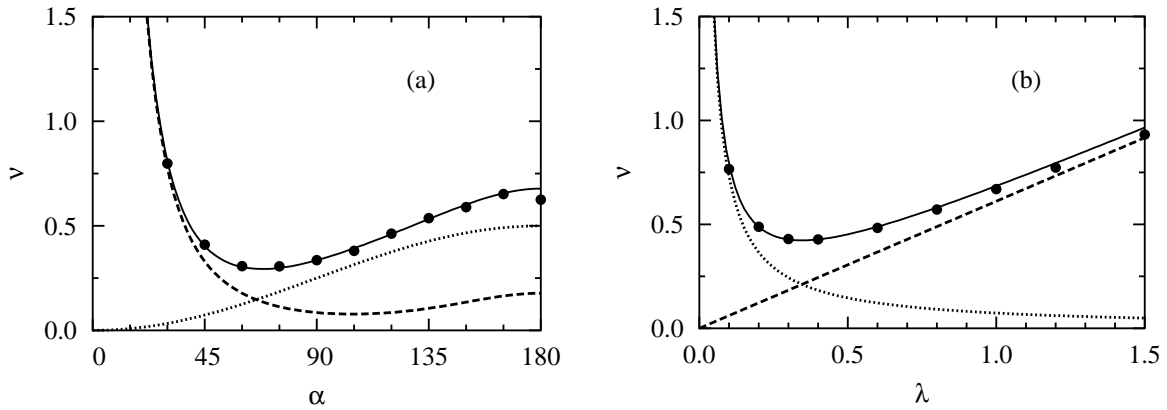


Fig. 8: Dimensionless kinematic viscosity for the simple fluid in MPC. Symbols are the simulation results, solid line is the total theoretical prediction, dotted line is the collisional contribution and dashed line the kinetic contribution. In both cases the system size is $L/a = 20$. In (a) α dependence is displayed with $\lambda = 0.2$ and $\rho = 10$. (b) shows the λ dependence with $\alpha = 130$ and $\rho = 5$.

The ratio between the kinetic and the collisional contributions to the kinematic viscosity varies considerably with the model parameters, as can be seen easily from the theoretical expressions (45). In Fig. 8 the total kinematic viscosity and its two contributions are plotted as a function of the rotation angle and the collision time step. The collisional contribution is dominant for large collision angles and small collision times, while the kinetic viscosity dominates in the opposite case of small collision angles and large collision times.

Kinetic transport is due to the movement of the particles themselves, *i.e.* when a particle moves it carries a certain amount of the relevant quantities as momentum and energy, while collisional transport is due to transfer of energy and momentum from one particle to another during collisions. In MPC, kinetic transport is therefore dominant when the mean free path is larger than the size of the collision box and for small values of the rotation angle. If the rotation angle is small, there is little exchange of momentum between particles due to collisions. The situation where the kinetic transport dominates is characteristic for gases. In fluids the usual situation is the opposite, the transport of momentum is mainly due to collisions.

Diffusion coefficient - The self-diffusion coefficient can be calculated by the Green-Kubo formalism from the velocity autocorrelation function (VACF) as $D = \frac{1}{3} \int_0^\infty dt \langle \mathbf{v}(t) \mathbf{v}(0) \rangle$ [82]. In order to obtain an analytical prediction, the Brownian approximation is employed for the VACF, what yields to,

$$D = \lambda \left(\frac{1}{\gamma} - \frac{1}{2} \right), \quad \gamma = \frac{2}{3} (1 - \cos \alpha) \left(1 - \frac{1}{\rho} \right), \quad (46)$$

where the decorrelation factor γ has been specified for the 3-dimensional case. A simulation measurement can be obtained through the VACF or by calculating the averaged mean square displacement. The agreement with simulation results is quite reasonable for large values of the mean free path λ , but not so for small ones. This is due to the fact that the Brownian approximation neglects the effect of correlations between particles. This deviation is therefore an indication of the importance of the enhancement of the collective behavior in the regime where the mean free path is smaller than the box size [79, 80].

Schmidt number - A convenient measure of the importance of hydrodynamics is the Schmidt number $Sc = \nu/D$, where ν is the kinematic viscosity and D the diffusion coefficient. Thus, Sc is the ratio between momentum transport and mass transport. It is known that this number for gases is on the order of unity, while in fluids like water it is on the order of 10^2 to 10^3 . A prediction for the Schmidt number of a MPC fluid can be obtained from the theoretical expressions (45) for the kinematic viscosity, and the diffusion coefficient in Eq. (46). In Fig. 9, the theoretical prediction is plotted for Sc as a function of the mean free path (or collision time) for different values of the rotation angle. This shows that Sc becomes considerably larger than unity for the same range of parameters where the collisional viscosity is considerably larger than the kinetic viscosity.

In order to perform simulations where the hydrodynamic effect are more easily taken into account, it is advisable to employ large values of the rotation angle α (like $\alpha = 130^\circ$) and small values of the collision time (like $h = 0.1$ or smaller). These restrictions, together with the election of the number density ρ , still give a large margin to chose particular values for the transport coefficients like the viscosity $\eta = \nu\rho$. However, it should be recall that the simulations will become computationally more expensive as for smaller collision times and for larger number densities.

4.3 Implementation of complex structures

After regarding the implementation and characteristic behavior of a simple fluid with simulated with MPC, the next important question is how complex fluids can be modeled. Two simple examples of complex structures are colloidal suspensions and polymer solutions. To this end, the

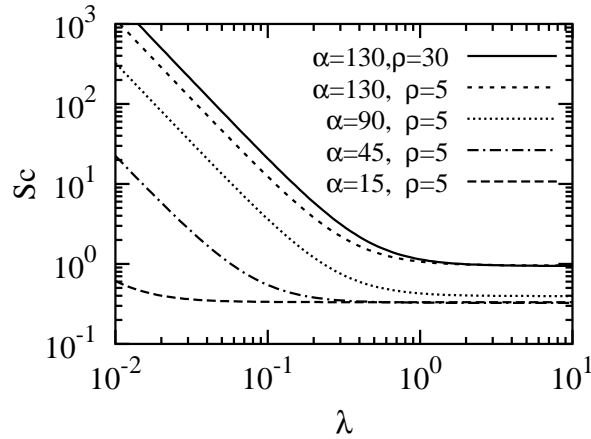


Fig. 9: Theoretical Schmidt number versus collision time. The α and ρ parameters are specified in the plot.

strategy is to define an hybrid algorithm where the solvent is simulated with the MPC technique and the solute particles are described with standard Molecular dynamics simulations(MD).

In this hybrid model the MPC streaming step in Eq. (38) is used to update the positions just of the solvent particles, while in the collision step in Eq. (40) both solvent and solute particles are taken into account. The collision is performed through the collision box center of mass velocity Eq. (39) where the solute particles are generally considered to have larger mass than the surrounding solvent particles. The position update of the solute particles is performed in several MD steps between MPC collisions, and it is in the MD where the particular characteristics of the interaction potential are taken into account.

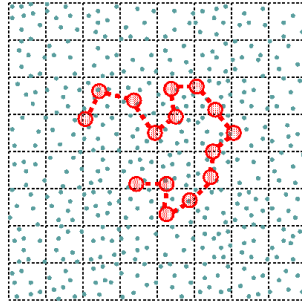


Fig. 10: Diagram of the inclusion of a polymer in a MPC solvent.

For example, in case a polymer is considered [83,84], N_m point particles of mass M are taken to be the polymer monomers. These monomers are connected by harmonic springs and eventually also interact via an excluded-volume potential (for details of these potentials see Chapter B3). Typical potential parameters are then related to the MPC units. The typical distance between consecutive monomers (bond length) could be equal to the box size, so that there is no more than one monomer in each collision box. The potential strength would similarly be taken in units of the thermal energy $k_B T$. The MD time steps are integrated, for instance, with the velocity-Verlet algorithm with a time step Δt generally 20 to 100 times smaller than the collision time.

In other words, we consider a system of monomers or colloidal particles interacting through particular potentials whose positions and velocities evolve in discrete time intervals Δt . This procedure is interrupted every $h/\Delta t$ steps for the interaction with the fluid particles. This

interaction is a MPC event where solvent and solute particles interchange momentum. This implies that the solvent particles can enter the cores of the colloidal particles, but the colloids cannot inter penetrate each other.

The hybrid model described here is a variant of the model introduced previously by Malevanets and Kapral [72, 85] for colloidal particles. In their model, both the solute-solute and solute-solvent interactions were taken into account through excluded-volume potentials with MD, and only the solvent-solvent interactions were mesoscopically described through MPC. The advantage of the model described here comes from the fact that in the MD steps just the solute particles are considered. This leads to a considerable speed up of the simulations.

An optimal choice for the mass of the solute particle in order to enhance the hydrodynamic coupling between solute and fluid particles is $M/m \simeq \rho$. This can be understood as follows. The relative mass of the solute and solvent particles appears in the collision step via the calculation of the center-of-mass velocity. If solute particles have the same mass as solvent particles and there is a large number of solvent particles per cell, the solvent particles transfer a large random momentum to the solute particle. Simultaneously, the effect of the solute particle momentum on the solvent is small. A very large mass of the solute particle is not very convenient either, because it implies a large ballistic regime and a long diffusion time (see more details in Ref. [80]).

These hybrid method has been used to study the dynamics of short polymer chains in solution [79, 84]. The predictions of the Zimm theory, in which the effect of hydrodynamic interactions is taken into account, are in excellent agreement with the simulation results for instance, for the center of mass diffusion coefficient and for the relaxation times of the Rouse modes. This agreement is found in the parameter regime where the collective behavior has been described to dominate the dynamics, which also is when the Schmidt number displayed larger values. Other studies [86, 87] have shown, by means of MPC simulations, the relevance of hydrodynamics interactions in two particular problems, polymer collapse and sedimenting colloidal suspensions respectively.

MPC arises therefore as a promising method in the broad field of mesoscale simulations. It is nevertheless a quite recent simulation method, such that there is still space for further improvements, and it has to be tested in several systems in comparison with previous techniques.

Acknowledges

I would like to thank Antonio Lamura and to Roland G. Winkler for reading the manuscript.

References

- [1] M. Karttunen, I. Vattulainen, and A. Lukkarinen, eds., *Novel Methods in Soft Matter Simulations* (Springer, 2004).
- [2] S. Succi, *The Lattice Boltzmann Equation: for fluid dynamics and beyond* (Clarendon, Oxford, 2001).
- [3] N. Attig, K. Binder, H. Grubmüller, and K. Kremer, eds., *Computational Soft Matter: From Synthetic Polymer to Proteins* (NIC-Directors, 2004).
- [4] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Academic Press, San Diego, 2002), 2nd ed.
- [5] W. W. Wood and F. R. Parker, *J. Chem. Phys.* **27**, 720 (1957).
- [6] L. B. Lucy, *Astron. J.* **82**, 1013 (1977).
- [7] J. J. Monaghan, *Annu. Rev. Astron. Astrophys.* **30**, 543 (1992).
- [8] H. A. Posch, W. G. Hoover, and O. Kum, *Phys. Rev. E* **52**, 1711 (1995).
- [9] W. G. Hoover and H. A. Posch, *Phys. Rev. E* **54**, 5142 (1996).
- [10] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics* (Pergamon Press, 1959).
- [11] G. A. Bird, *Molecular Gas Dynamics* (Clarendon, Oxford, 1976).
- [12] F. J. Alexander, A. L. Garcia, and B. J. Alder, *Phys. Rev. Lett.* **74**, 5212 (1995).
- [13] J. Hardy, Y. Pomeau, and O. de Pazzis, *J. Math. Phys.* **14**, 1746 (1973).
- [14] U. Frisch, B. Hasslacher, and Y. Pomeau, *Phys. Rev. Lett.* **56**, 1505 (1986).
- [15] A. J. C. Ladd, M. E. Colvin, and D. Frenkel, *Phys. Rev. Lett.* **60**, 975 (1988).
- [16] J. M. V. A. Koelman, *Phys. Rev. Lett.* **64**, 1915 (1990).
- [17] G. R. McNamara and G. Zanetti, *Phys. Rev. Lett.* **61**, 2332 (1988).
- [18] X. He and L. S. Luo, *Phys. Rev. E* **56**, 6811 (1997).
- [19] S. Chen and G. D. Doolen, *Ann. Rev. Fluid Mech.* **30**, 329 (1998).
- [20] R. R. Nourgaliev, T. N. Dinh, T. G. Theofanous, and D. Joseph, *Int. J. Multiphase Flow.* **29**, 117 (2003).
- [21] Y. Qian, D. d’Humières, and P. Lallemand, *Europhys. Lett.* **17**, 479 (1992).
- [22] A. J. C. Ladd, *J. Fluid Mech.* **271**, 285 (1994).
- [23] A. J. C. Ladd, H. Gang, J. X. Zhu, and D. A. Weitz, *Phys. Rev. E* **52**, 6550 (1995).
- [24] P. J. Hoogerbrugge and J. M. V. A. Koelman, *Europhys. Lett.* **19**, 155 (1992).

- [25] C. Marsh, G. Backx, and M. Ernst, *Phys. Rev. E* **56**, 1976 (1997).
- [26] C. Marsh, G. Backx, and M. Ernst, *Europhys. Lett.* **38**, 411 (1997).
- [27] J. Bonet-Avalós and A. D. Mackie, *Europhys. Lett.* **40**, 141 (1997).
- [28] P. Español, *Europhys. Lett.* **40**, 631 (1997).
- [29] H. Risken, *The Fokker Planck Equation* (Springer, Berlin, 1989).
- [30] C. W. Gardiner, *Handbook of Stochastic Methods* (Springer, Berlin, 1983).
- [31] P. Español and P. Warren, *Europhys. Lett.* **30**, 191 (1995).
- [32] P. Español and M. Revenga, *Phys. Rev. E* p. 026705 (2003).
- [33] P. Español, M. Serrano, and H. C. Öttinger, *Phys. Rev. Lett.* **83**, 4552 (1999).
- [34] I. Pagonabarraga and D. Frenkel, *J. Chem. Phys.* **115**, 5015 (2001).
- [35] C. Marsh and J. M. Yeomans, *Europhys. Lett.* **37**, 511 (1997).
- [36] R. D. Groot and P. B. Warren, *J. Chem. Phys.* **107**, 4423 (1997).
- [37] J. B. Gibson, K. Chen, and S. Chynoweth, *Int. J. of Mod. Phys. C* **10**, 241 (1999).
- [38] I. Pagonabarraga, M. H. J. Hagen, and D. Frenkel, *Europhys. Lett.* **42**, 377 (1998).
- [39] G. Besold, I. Vattulainen, M. Karttunen, and J. M. Polson, *Phys. Rev. E* **62**, R7611 (2000).
- [40] W. K. den Otter and J. H. R. Clarke, *Europhys. Lett.* **53**, 426 (2001).
- [41] T. Shardlow, *SIAM J. Sci. Compt.* **24**, 1267–1282 (2003).
- [42] P. Nikunen, M. Karttunen, and I. Vattulainen, *Comput. Phys. Commun.* **153**, 407 (2003).
- [43] G. D. Fabritiis, M. Serrano, P. Español, and P. V. Coveney, *Physica A* **361**, 429–440 (2006).
- [44] M. Ripoll, M. H. Ernst, and P. Español, *J. Chem. Phys.* **115**, 7271 (2001).
- [45] J. M. V. A. Koelman and P. J. Hoogerbrugge, *Europhys. Lett.* **21**, 363 (1993).
- [46] E. S. Boek and P. van der Schoot, *Int. J. of Mod. Phys. C* **9**, 1307 (1998).
- [47] W. Dzwinel and D. A. Yuen, *Int. J. of Mod. Phys. C* **11**, 1037 (2000).
- [48] M. Whittle and E. Dickinson, *J. Colloid and Interface Science* **242**, 106 (2001).
- [49] A. G. Schlijper, P. J. Hoogerbrugge, and C. W. Manke, *J. Rheol.* **39**, 567 (1995).
- [50] Y. Kong, C. W. Manke, W. G. Madden, and A. G. Schlijper, *J. Chem. Phys.* **107**, 592 (1997).
- [51] N. A. Spenley, *Europhys. Lett.* **49**, 534 (2000).

- [52] A. G. Schlijper, C. W. Manke, W. D. Madden, and Y. Kong, *Int. J. of Mod. Phys. C* **8**, 919 (1997).
- [53] K. E. Novik and P. V. Coveney, *Phys. Rev. E* **61**, 435 (2000).
- [54] S. I. Jury, P. Bladon, S. Krishna, and M. E. Cates, *Phys. Rev. E* **59**, R2535 (1999).
- [55] V. M. Kendon, J. C. Desplat, P. Bladon, and M. E. Cates, *Phys. Rev. Lett.* **83**, 576 (1999).
- [56] C. M. Wijmans, B. Smit, and R. D. Groot, *J. Chem. Phys.* **114**, 7644 (2001).
- [57] S. M. Willemsen, T. J. H. Vlugt, H. C. J. Hoefsloot, and B. Smit, *J. Comp. Phys.* **147**, 507 (1998).
- [58] R. D. Groot and T. J. Madden, *J. Chem. Phys.* **108**, 8713 (1998).
- [59] R. D. Groot, T. J. Madden, and D. J. Tildesley, *J. Chem. Phys.* **110**, 9739 (1999).
- [60] J. Jones, M. Lal, N. Ruddock, and N. A. Spenley, *Faraday Discussions* **112**, 129 (1999).
- [61] P. Malfreyt and D. Tildesley, *Langmuir: ACS journal of surfaces and colloids* **16**, 4732 (2000).
- [62] S. I. Jury, P. Bladon, M. E. Cates, S. Krishna, M. Hagen, N. Ruddock, and P. B. Warren, *Phys. Chem. Chem. Phys.* **1**, 2051 (1999).
- [63] M. Venturoli and B. Smit, *Phys. Chem. Comm.* **10**, 45 (1999).
- [64] J. A. Elliott and A. H. Windle, *J. Chem. Phys.* **113**, 10367 (2000).
- [65] M. Ripoll, P. Español, and M. H. Ernst, *Int. J. of Mod. Phys. C* **9**, 1329 (1998).
- [66] M. Ripoll and M. H. Ernst, *Phys. Rev. E* **71**, 041104 (2005).
- [67] M. Ripoll, *Kinetic theory of dissipative particle dynamics models*, Ph.D. thesis, Universidad Nacional de Educación a Distancia, Spain (2002), (unpublished).
- [68] J. Bonet-Avalós and A. D. Mackie, *J. Chem. Phys.* **11**, 5267 (1999).
- [69] P. Español, *Phys. Rev. E* **57**, 2930 (1998).
- [70] B. I. M. ten Bosch, *J. Non-Newtonian Fluid Mech.* **83**, 231 (1999).
- [71] A. Malevanets and R. Kapral, *J. Chem. Phys.* **110**, 8605 (1999).
- [72] A. Malevanets and R. Kapral, *J. Chem. Phys.* **112**, 7260 (2000).
- [73] E. Tüzel, M. Strauss, T. Ihle, and D. M. Kroll, *Phys. Rev. E* **68**, 036701 (2003).
- [74] E. Allahyarov and G. Gompper, *Phys. Rev. E* **66**, 036702 (2002).
- [75] T. Ihle and D. M. Kroll, *Phys. Rev. E* **63**, 020201(R) (2001).
- [76] T. Ihle and D. M. Kroll, *Phys. Rev. E* **67**, 066705 (2003).

- [77] A. Lamura, G. Gompper, T. Ihle, and D. M. Kroll, *Europhys. Lett.* **56**, 319 (2001).
- [78] D. J. Tritton, *Physical Fluid Dynamics* (Oxford Science publications, 1988), 2nd ed.
- [79] M. Ripoll, K. Mussawisade, R. G. Winkler, and G. Gompper, *Europhys. Lett.* **68**, 106 (2004).
- [80] M. Ripoll, K. Mussawisade, R. G. Winkler, and G. Gompper, *Phys. Rev. E* **72**, 016701 (2005).
- [81] N. Kikuchi, C. M. Pooley, J. F. Ryder, and J. M. Yeomans, *J. Chem. Phys.* **119**, 6388 (2003).
- [82] T. Ihle and D. M. Kroll, *Phys. Rev. E* **67**, 066706 (2003).
- [83] A. Malevanets and J. M. Yeomans, *Europhys. Lett.* **52**, 231 (2000).
- [84] K. Mussawisade, M. Ripoll, R. G. Winkler, and G. Gompper, *J. Chem. Phys.* **123**, 144905 (2005).
- [85] A. Malevanets and R. Kapral, *Lec. Notes Phys.* **640**, 116 (2004).
- [86] N. Kikuchi, A. Gent, and J. M. Yeomans, *Eur. Phys. J.* **9**, 63 (2002).
- [87] J. T. Padding and A. A. Louis, *Phys. Rev. Lett.* **93**, 220601 (2004).