Self-consistent dissipative particle dynamics algorithm

I. Pagonabarraga, M. H. J. Hagen and D. Frenkel

FOM Institute for Atomic and Molecular Physics
Kruislaan 407, 1098 SJ Amsterdam, The Netherlands

Abstract. – We propose an implementation of dissipative particle dynamics that is free of the inconsistencies that plagued earlier algorithms. The present algorithm satisfies a form of microscopic reversibility. As a consequence, we recover the correct equilibrium properties. Moreover, we can use much larger time steps than previously. We report a detailed comparison between simulated transport properties and the theoretical predictions. We find that the existing theory is only valid under very special conditions. A more general theory is still lacking.

Fully atomistic Molecular Dynamics (MD) simulations are essentially useless for the study of the dynamics of complex liquids —the relevant length and time scales are such that the computational requirements exceed the capabilities of present-day supercomputers by several orders of magnitude.

During the past few years, a number of "coarse-grained" simulation techniques have been introduced to overcome this problem. In the context of colloidal systems and porous media, much progress has been made using lattice-Boltzmann simulations [1]. Both this method and the closely related lattice-gas cellular automaton model [2] have the drawback that they are defined on a lattice. This creates several problems, as the lattice lacks the full isotropy of the original system and does not satisfy Galilean invariance. Special care must be taken to correct this problem. Finally, lattice-gas models are ill-suited to model complex macromolecular systems, such as polymer solutions.

A model that does not suffer from these drawbacks is the dissipative particle dynamics method (DPD) introduced by Hoogerbrugge and Koelman in 1992 [3]. In this method particle positions and velocities are continuous, and their values are determined by knowledge of the forces acting on the particles. In this sense, the model is similar to MD. However, the DPD particles need not correspond to real particles —in some cases they can be interpreted as representations of fluid elements. The interactions between these "particles" are not conservative. If we denote the positions and momenta of particles $i = 1, \ldots, N$ at time t

by $(\mathbf{r}_i(t), \mathbf{p}_i(t))$ then the DPD equations of motion are

$$\frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t} = \mathbf{v}_i(t)\,, \tag{1}$$

$$\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = \mathbf{f}_i(t) \equiv \sum_{j \neq i} \left(\mathbf{F}^{\mathrm{C}}(\mathbf{r}_{ij}) + \mathbf{F}^{\mathrm{D}}(\mathbf{r}_{ij}, \mathbf{v}_{ij}) + \mathbf{F}^{\mathrm{R}}_{ij}(\mathbf{r}_{ij}) \right)\,. \tag{2}$$

The velocities $\mathbf{v}_i$ are defined as $\mathbf{p}_i/m_i$, where m_i is the mass of particle i. In the above equation we have used the notation $\mathbf{r}_{ij} \equiv \mathbf{r}_i - \mathbf{r}_j$ and $\mathbf{v}_{ij} \equiv \mathbf{v}_i - \mathbf{v}_j$. All the forces have an interaction range r_{c}. $\mathbf{F}^{\mathrm{C}}_{ij}$ stands for the conservative force that particle j exerts on i. The second term, $\mathbf{F}^{\mathrm{D}}_{ij}$, corresponds to a frictional force. It depends both on the relative positions and velocities of the particles, $\mathbf{F}^{\mathrm{D}}_{ij} \equiv -\gamma\omega^{\mathrm{D}}(r_{ij})\mathbf{v}_{ij} \cdot \hat{r}_{ij}\hat{r}_{ij}$ ($\hat{r}$ denotes a unit vector in the direction of $\mathbf{r}$). γ is a coefficient that controls the strength of the frictional force between DPD particles, while $\omega^{\mathrm{D}}(r)$ describes the variation of the friction with distance. Finally, a random force, $\mathbf{F}^{\mathrm{R}}_{ij}$, is introduced. It depends only on the relative positions, $\mathbf{F}^{\mathrm{R}} \equiv \sigma\omega^{\mathrm{R}}(r_{ij})\hat{r}_{ij}\xi_{ij}$. σ determines the magnitude of the random pair-force and ω^{R} its distance dependence. Here ξ_{ij} is a random variable with Gaussian distribution and unit variance. To enforce momentum conservation, we require that $\xi_{ij} = \xi_{ji}$. The presence of dissipative forces makes this model similar to Brownian dynamics. But as all forces obey action-equals-reaction, the model conserves momentum. This feature is essential to recover "hydrodynamic" (Navier-Stokes) behaviour on sufficiently large length and time scales.

The relation between DPD and statistical mechanics was investigated by Español and Warren [4], who noticed that, in order to recover the proper thermodynamic equilibrium for a DPD fluid at a prescribed temperature T, it was necessary to impose a fluctuation-dissipation relation between the dissipative and the random forces. They showed that the probability to find the system in a given spatial configuration is given by the Boltzmann weight of that state only if $\omega^{\mathrm{D}}(r_{ij}) = \omega^{\mathrm{R}}(r_{ij})^2$ and $\sigma^2 = 2k_{\mathrm{B}}T\gamma$, with k_{B} the Boltzmann constant. The relation between dissipative particle dynamics and hydrodynamics was investigated by Español and by Marsh *et al.* [5], [6] who showed that the DPD model should obey the Navier-Stokes equation, at least in the limit where the time step tends to zero. In real numerical simulations, the equations of motion (1) are integrated using a finite time step. As the DPD algorithm satisfies conservation of mass and momentum, it seems likely that the finite time step version will also exhibit the correct hydrodynamic behaviour. Approximate expressions for the transport coefficients of the DPD fluid have only been derived in the limit that the time step of the discretized equations of motion tends to zero [6].

The fact that the DPD model reproduces Boltzmann statistics and Navier-Stokes behaviour makes it an attractive numerical tool to simulate complex fluids. Indeed, several authors [7], [8] have already applied the method for this purpose.

Yet, although *in principle* everything is correct, *in practice* the DPD model is still plagued by serious problems. This is most obvious in the case of the dissipative ideal gas where no conservative interactions act between the particles. For this model system, it has been observed that the equilibrium properties depend on the magnitude of the time step used in the simulation [9]. For instance, the temperature measured in simulations depends on the time step and differs in general from the value given by the fluctuation-dissipation relation. Moreover, the radial-distribution function which, for an ideal gas, should be flat, is found to exhibit a pronounced structure that depends on the time step [6]. In addition, it is still unclear under what (if any) conditions the approximate theoretical predictions for the transport properties of the DPD model become exact. The available numerical data [6], [10] show that the theoretical predictions for the self-diffusion coefficient and the viscosity of a DPD fluid,

although qualitatively correct, are at best fair, when a quantitative comparison is performed.

In this letter, we show that a proper choice of the numerical algorithm makes it possible to reproduce the correct equilibrium behaviour of the DPD model. With this algorithm, we then study the transport properties of the fluid and compare to the theoretical predictions.

The standard implementation of eqs. (1), (2) makes use of the Euler scheme, *i.e.*, the new velocities and positions at time $t + \Delta t$ are derived from the knowledge of the same quantities at time t [3]. In conventional MD, such algorithms are avoided because they lead to a drift in the energy of the system. This problem seems less serious in DPD, because the combination of dissipative and random forces is chosen such that the temperature should remain constant. However, the problem that Euler-type algorithms are not time reversible remains. As the DPD algorithm is stochastic, we should explain what we mean by time reversibility. We require that, for a given time sequence of the random numbers that are used to generate the random forces, all trajectories should be strictly reversible. This is not true for Euler-type algorithms. However, in MD simulations, it is well known that a Verlet-style algorithm, such as the leap-frog scheme, satisfies time reversibility of the trajectories for any size of the time step (see, *e.g.*, [11]). In the leap-frog scheme, the velocities are defined at half the time interval, and their values for the next time are evaluated on the basis of the knowledge of the positions at the previous time step. The velocities and positions are then updated as follows:

$$\mathbf{v}_i \left(t + \frac{\Delta t}{2} \right) = \mathbf{v}_i \left(t - \frac{\Delta t}{2} \right) + \Delta t \frac{\mathbf{f}_i(t)}{m} , \tag{3}$$

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \mathbf{f}_i(t) \frac{(\Delta t)^2}{2m} . \tag{4}$$

The new feature in the implementation of the leap-frog algorithm in DPD is that the force $\mathbf{f}_i(t)$ acting on particle i depends on the velocities $\{\mathbf{v}_j(t)\}$ of neighbouring particles. The velocities at the intermediate time steps are given by $\mathbf{v}(t) = \frac{1}{2}(\mathbf{v}(t - \Delta t/2) + \mathbf{v}(t + \Delta t/2))$. This means that the force term in eq. (3) depends linearly on $\mathbf{v}_i(t - \Delta t/2)$ and $\mathbf{v}_i(t + \Delta t/2)$. Since the updated velocity of particle i will depend on the new velocities of the neighbouring particles, the updating of eq. (3) has to be performed self-consistently, *i.e.* the final values of the forces and velocities have to be consistent. This algorithm is essentially different from the Verlet-type algorithms that have been introduced in the literature [8] as a means of improving the initial Euler scheme for DPD. These algorithms are more accurate, but time reversibility is still not preserved. Most importantly, with the self-consistent scheme the maximum attainable Δt is no longer limited by the algorithm. This more than compensates the fact that this algorithm is typically twice as expensive as the original one.

Let us analyze what time-reversibility implies for detailed balance. To this end, we consider a single DPD time step as a Monte Carlo move. For the sake of simplicity, we consider the displacement of a single particle i. In the Euler algorithm, the probability to move this particle from $\mathbf{r}_i$ to $\mathbf{r}_i + \Delta \mathbf{r}$ is determined by the force acting on that particle at its initial position $\mathbf{r}_i$. The probability to carry out the reverse move from $\mathbf{r}_i + \Delta \mathbf{r}$ to $\mathbf{r}_i$ depends on the force acting on the particle at $\mathbf{r}_i + \Delta \mathbf{r}$. This force will, in general, be different, and hence detailed balance is not satisfied. In contrast, for the self-consistent algorithm, the update rule for the forward and reverse moves is the same since the algorithm is time reversible. If only dissipative forces act between the particles, this algorithm satisfies detailed balance. In the presence of conservative forces, small deviations from detailed balance may occur, but this is due to the fact that, just like conventional MD, the algorithm works with forces rather than potential energies. And, as in MD, this will lead to small (and usually harmless) discretization errors.

To test the present scheme, we have performed numerical simulations on a dissipative ideal gas. The reason to select this model is twofold: first of all, the difference between the present
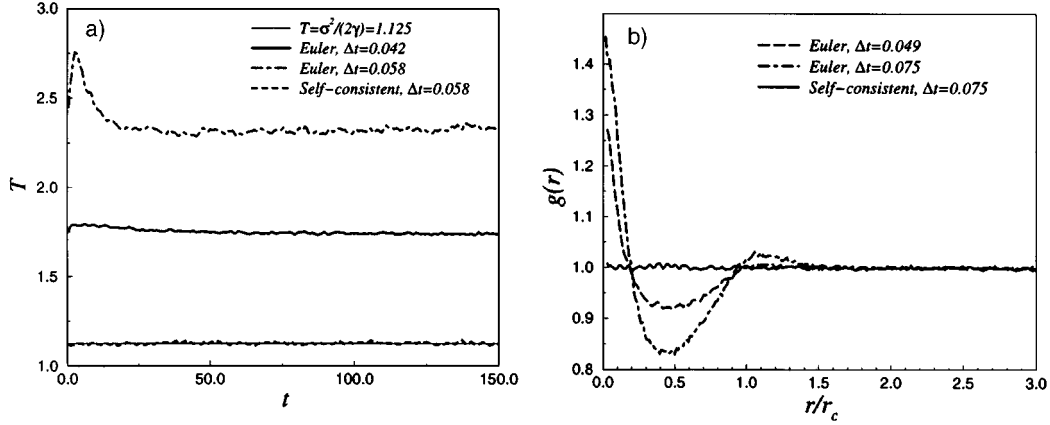
Fig. 1. – Comparison of the equilibrium properties of an ideal 2D DPD gas with the self-consistent and the Euler algorithms, for $\sigma=1.5$, $\gamma=1$, $n=25$, $L=10r_c$, $r_c=4$. a) Measured and imposed temperature; t is the time in units of $r_c\sqrt{m/(2k_BT)}$. b) Radial distribution function.

approach and the Euler scheme should be easy to detect for this model. In addition, we can compare the theoretical predictions for the viscosity of this model with our numerical data. Here, we limit the discussion to the two-dimensional dissipative ideal gas. We assume that all particles have unit mass and that ω^D is given by $\omega^D(r) = 2(1-r/r_c)$.

Let us first consider the equilibrium properties of the model system. As no conservative forces act between the particles, the system should have the same equilibrium properties as an ideal gas. In particular, the fluid should be free of structural correlations and the temperature should equal the temperature that is imposed through the fluctuation-dissipation relation. Figure 1a) shows a comparison of the temperature obtained using the self-consistent and the Euler algorithms. In the first case, one can see that, throughout the simulation, the "measured" temperature coincides with the imposed value ($T = \sigma^2/(2\gamma) = 1.125$ —there is a small finite-size correction that is not visible at the scale of the figure). The temperature obtained using the Euler algorithm first exhibits a transient and then settles on the wrong value. Due to the lack of detailed balance, the system has an effective temperature that differs from the imposed value (see ref. [9]). Clearly, both the transient and the limiting behaviour of the temperature are due to the use of the Euler algorithm and not to any intrinsic problems with the DPD scheme reaching thermal equilibrium [9].

Next, we consider $g(r)$, the radial distribution function of the DPD gas. As there are no static correlations, we expect $g(r) = 1$ for all r. However, Marsh et al. [6] reported that $g(r)$ tends to develop a pronounced structure as the time step Δt increases. In fig. 1b) we compare the predictions of the two algorithms. The figure clearly illustrates that the structure in $g(r)$ is an artifact of the Euler algorithm. The lack of detailed balance implicit in the algorithm leads to unphysical correlations between the particle positions. The behaviour of a DPD fluid close to a solid wall is also of practical relevance. In equilibrium one expects a uniform density profile up to the boundary itself. As with $g(r)$, we find that the Euler scheme leads to spurious structure that dissapears with the self-consistent algorithm [12]. The ability to avoid structuring of a DPD fluid close to a wall is important when modelling colloidal suspensions as structuring of the fluid will induce spurious depletion forces between the colloids.

Next, we turn to the transport properties of the DPD fluid. We focus on the shear viscosity. We consider a system undergoing uniform shear flow. Earlier (Euler-algorithm) simulations
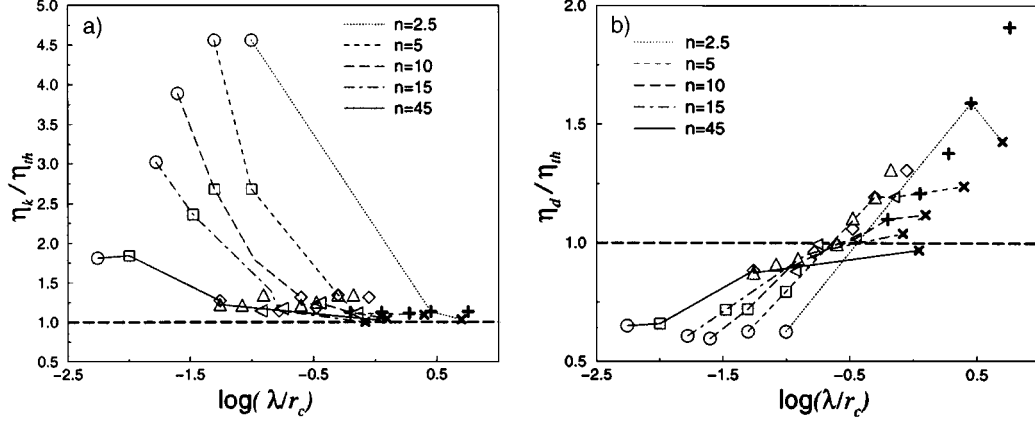
Fig. 2. – Kinetic and dissipative contributions to the shear viscosity in an ideal 2D DPD fluid as a function of λ/r_c. ○: $T = 0.5$, $r_c = 4, \gamma = 1$, □: $T = 2$, $r_c = 4$, $\gamma = 1$, ◇: $T = 50$, $r_c = 4$, $\gamma = 1$, × : $T = 0.5$, $r_c = 4$, $\gamma = 0.1$, +: $T = 1$, $r_c = 4$, $\gamma = 0.1$, ◁: $T = 1$, $r_c = 2$, $\gamma = 0.1$, △: $T = 1250$, $r_c = 4$, $\gamma = 1$. a) Kinetic contribution. $\eta_{\rm th}$ is the theoretical prediction of ref. [6], $\eta_{\rm th} = 3T/(2\pi\gamma r_c^2)$. b) Interaction contribution. In this case, $\eta_{\rm th} = \gamma n^2/(80\pi)$. System sizes range between 10 and 25 r_c, depending on the parameters.

of Marsh *et al.* found that there was a discrepancy between the shear viscosities predicted theoretically and the numerical results [6]. As the theory of ref. [6] is approximate, this observation is, in itself, not worrying. However, it is important to rule out any artifacts due to the algorithm. In our simulations, we have imposed Lees-Edwards boundary conditions [13] to simulate Couette flow without solid boundaries. In conventional MD, these boundary conditions only play a role when updating the particle coordinates and velocities. In DPD, the moving boundary conditions also affect the computation of $\mathbf{F}_{ij}^{\rm D}$. As we are interested in the Newtonian regime, we should take care that the imposed shear is sufficiently small. For DPD, where the temperature sets a natural energy scale, we have always applied shear rates such that its maximum velocity is at most of the order of the thermal velocity.

As in atomic fluids, one can distinguish two contributions to the stress tensor. One has a kinetic origin: it is due to the motion of the DPD particles themselves. This part is responsible for the kinetic contribution to the shear viscosity, η_k. The second contribution to the viscosity, η_d, is associated with momentum transport due interaction between the DPD particles. Note that for DPD this second contribution also (and, in the case of an ideal gas, only) depends on the velocities of the particles. The theory predicts that the random forces do not contribute to the viscosity, and this is indeed what we find in our simulations.

There are two relevant length scales in the DPD model. One is r_c. The second is the distance a DPD particle can travel before its velocity is randomized. For simple gases this distance is proportional to the mean free path between successive collisions. However, when dissipative forces dominate the decay of velocity fluctuations, this characteristic length is inversely proportional to the friction coefficient and to the total number of interacting particles. In a two-dimensional DPD fluid we define it as $\lambda \equiv \sqrt{2T}/(n\gamma)$. $n = N\pi r_c^2/L^2$ is the average number of interacting particles, with N the total number of particles and L the side of the system. When $\lambda > r_c$, one expects that a DPD particle will interact with different particles on successive collisions.

Figure 2 shows a comparison of our simulation results for the the kinetic and dissipative parts of the viscosity with the theoretical predictions [6]. In this figure, we have collected simulation results for a wide range of values of σ and γ (and therefore temperatures). Our

data differ appreciably from the earlier simulations of ref. [6], but the discrepancy with theory remains, in particular for the kinetic contribution to the viscosity, although relative deviations of the total η may be smaller due to fortuitous cancellation of errors. The discrepancy between theory and simulation appears to be least serious for high values of the ratio λ/r_c. This can be expected, because the theory of ref. [6] makes the assumption that collisions are instantaneous. This approximation becomes better as $\lambda/r_c \to \infty$. The theory also predicts that η_k does not depend on the density. The simulations show that, certainly when $\lambda < r_c$, this is not true. In atomic fluids, such a density dependence of η_k is due to recollisional effects —in DPD the corresponding phenomenon is that two particles can interact many times before they move apart, which happens when $\lambda \ll r_c$. In fig. 2 we have joined values which correspond to the same number of average interacting particles, n. As n increases, the deviations from the theoretical predictions become smaller for all λ/r_c. Therefore, the (mean field) theory works better for large n, as is to be expected.

For practical applications of DPD, it is important to know at what time scales hydrodynamic behaviour sets in. To this end, we have studied the decay of an imposed shear wave. In the hydrodynamic regime, such a shear wave should decay exponentially. We have observed, however, that when $\lambda > r_c$, the initial decay is clearly not linear. In this limit the motion of the DPD particles is nearly ballistic. In contrast, when $\lambda < r_c$, exponential decay sets in very quickly [12]. Clearly, if the aim is to use DPD to model the hydrodynamic behaviour of complex fluids, then it is advisable to work in this regime. But, as yet, there is no good kinetic theory for the regime where $\lambda < r_c$.

$$***$$

REFERENCES

[1] Ladd A.J.C., *J. Fluid Mech.*, **271** (1994) 285.
[2] Frisch U., Hasslacher B. and Pomeau Y., *Phys. Rev. Lett.*, **56** (1986) 1505.
[3] Hoogerbrugge P. J. and Koelman J. M. V. A., *Europhys. Lett.*, **19** (1992) 155.
[4] Español P. and Warren P. B., *Europhys. Lett.*, **30** (1995) 191.
[5] Español P., *Phys. Rev. E*, **53** (1996) 1572.
[6] Marsh C. A., Backx G. and Ernst M. H., *Europhys. Lett.*, **38** (1997) 411; *Phys. Rev. E*, **56** (1997) 1676.
[7] Koelman J. M. V. A. and Hoogerbrugge P. J., *Europhys. Lett.*, **21** (1993) 363; Schlijper A. G., Hoogerbrugge P. J. and Manke C. W., *J. Rheol.*, **39** (1995) 567 Boek E. S., Coveney P. V., Lekkerkerker H. N. W. and Van Der Schoot P., *Phys. Rev. E*, **55** (1997) 3124. Coveney P. V. and Novik K. E., *Phys. Rev. E*, **54** (1996) 5134.
[8] Groot R. D. and Warren P. B., *J. Chem. Phys*, **107** (1997) 4423.
[9] Marsh C. A. and Yeomans J. M., *Europhys. Lett.*, **37** (1997) 511.
[10] Warren P. B., preprint.
[11] Frenkel D. and Smit B., *Understanding Molecular Simulation* (Academic Press, San Diego) 1996.
[12] Pagonabarraga I. and Frenkel D., to be published.
[13] Allen M. P. and Tildesley D. J., *Computer Simulation of Liquids* (Oxford University Press, Oxford) 1987.