Development of an automated reliable method to compute transport properties from DPD equilibrium simulations: Application to simple fluids

Development of an automated reliable method to compute transport properties from DPD equilibrium simulations: Application to simple fluids.

N. Lauriello[a], G. Boccardo[a], D. Marchisio[a], M. Lisal[b,c], A. Buffo[a]

[a]*DISAT – Institute of Chemical Engineering, Politecnico di Torino, C.so Duca degli Abruzzi 24, Torino, 10129, Italy*
[b]*Research Group of Molecular and Mesoscopic Modelling, The Czech Academy of Sciences, Institute of Chemical Process Fundamentals, Prague, Czech Republic*
[c]*Department of Physics, Faculty of Science, J. E. Purkinje University, Ústí n. Lab, Czech Republic*

Abstract

Dissipative particle dynamics (DPD) is a promising candidate technique for modeling rheological properties of soft matter systems. However, several methodological issues inhibit its exploitation as a computational rheology tool. In this work, we focus on the development of an automated method to compute transport properties from equilibrium simulation with particular attention to the assessment of the Green-Kubo approach reliability and computational feasibility for a large set of simple DPD systems with increasing Schmidt number. Furthermore, we investigate the time step size dependency of dynamic properties and the role of different time integration schemes. In particular, we assess the performance of the Shardlow-splitting algorithm against the most popular modified velocity-Verlet algorithm. We consider, for the first time, application of the Shardlow-splitting algorithm to the transverse DPD thermostat in different friction regimes relying on systematic numerical experiments. In addition, we make use of these findings to perform a multi-parametric study aiming to investigate the Schmidt number relationship with the effective friction coefficient.

Keywords: Automated algorithm, Dissipative Particle Dynamics, Green-Kubo formula, Equilibrium methods, Transport Properties, Transverse DPD thermostat

1. Introduction

It is now widely accepted that modern chemical engineering relies on computational methods for investigating the scale-up and design of equipment for the processing of soft matter materials, as they contribute to reduce the number of experiments [1]. Although multiscale modelling is a particularly challenging problem for this class of materials, it is the key to success in modelling their complex behaviour [2, 3, 4, 5]. Soft materials are very common in food, personal-care and pharmaceutical industries and include polymeric solutions and melts, suspension of colloidal particles, micellar solutions and liquid foams. Their rheological properties are very important as they define their possible applications, and they play a crucial role in the definition of design and scale-up rules for the corresponding processes and equipment. Their prediction requires the fluid characterization at lower scales than macro-scale. On the other hand, conventional molecular simulation methods would require an unfeasible simulation time because the characteristic size of the system range from nanometres to micrometres and the time evolution of the process under investigation can be of order of microseconds or milliseconds [6].

To overcome this limitation, a mesoscopic simulation technique able to bridge the gap between micro-scale and macro-scale models such as dissipative particle dynamics (DPD) is needed [7]. In DPD, groups of atoms and molecules are clustered into single units, so-called *beads*, each representing a complex molecular component of the system. *Coarse-graining* (CGing) plays a vital role in the DPD as it allows to simulate the hydrodynamic behaviour of complex fluids with a considerably lower number of particles, thereby significantly impacting on the speed of the simulation. Although CGing is a central element due its scale-bridging role, also other positive unique features make DPD very promising from the applications perspective, among mesoscopic simulation techniques. The quality and softness of effective interactions between beads allow to treat both static and flowing soft matter systems, at physically interesting length- and time-scales. It is considered one of the most flexible simulation techniques, widely used for soft matter systems, with applications to colloidal particles [8], polymer molecules [9] and fluid mixtures, self-assembly of surfactants [10] and polymers [11, 12, 13, 14], modeling of binary mixtures [15] or mixing in structured fluids [16]. Such materials have a complex rheological behaviour resulting from their morphological properties. Even though the rheological behaviour of complex fluids can be related to the structural changes occurring under fluid motion, it is important at first to estimate the viscosity of morphologies in equilibrium without perturbing

them. In fact, while the DPD ability to rapidly scan conformational space is well known, some issues related to the confident use of equilibrium approaches for the viscosity calculation remains unsolved [17, 18].

The viscosity resulting from equilibrium simulation is usually calculated by integrating the stress autocorrelation function (SACF), according to the Green-Kubo (GK) relation [19]. The main drawback of this approach is related to the large fluctuations associated with the stress tensor elements values, leading to poor accuracy in the resulting viscosity [20]. Therefore very long simulations are required to ensure good statistics in the SACF. Moreover, the direct use of the GK formula is often impossible since the calculation of the SACF will require the treatment of all the particle velocities at every investigated time step. Therefore, approximate algorithms based on different simplified strategies are commonly available in the main DPD codes. The influence of the adopted strategy in the calculation of the SACF is often overlooked, requiring an in-depth analysis, often carried out manually through a trial and error procedure, to obtain a final viscosity value independent on the simplified strategy.

In DPD, uncertainty in the results is also linked to the dissipative force, where the error in viscosity increases with increasing its importance as in the case of high Schmidt number fluids simulation. Therefore, special attention was devoted to the solution of some methodological issues related to the simulation of high Schmidt number fluids. In particular, this work investigated the role of different integration algorithms and the effect of time step size on the statistical accuracy using an extension of the standard DPD [21, 22, 23]. Another important aspect to consider is the capability of DPD to simulate high Schmidt number fluids. Indeed, it was claimed that although standard DPD [24, 25] gives good prediction of equilibrium static properties, it fails in reproducing the Schmidt number (Sc) of real fluids [7]. Two important transport properties are grouped into the Schmidt number, Sc, defined as the ratio between the fluid kinematic viscosity, ν, and the self-diffusion coefficient, $\mathscr{D}$:

$$Sc = \frac{\nu}{\mathscr{D}}. \tag{1}$$

The Sc characterizes the dynamic behaviour of fluids and it is an important quantity to consider in simulating unsteady and complex flows, since the Schmidt number of the solvent strongly affects non-equilibrium quantities of complex fluids, such as viscosity [26].

The soft potentials used in DPD do not transport momentum as efficiently as real interparticle potentials. In a liquid, unlike a gas, momentum can be trans-

ported rapidly by the intermolecular forces. On the other hand, mass transport occurs by the displacement of molecules and it is, by comparison, slow. Sc is therefore a large number (for water Sc is $O(10^3)$). The main drawback of standard DPD is the lack of shearing dissipation, which has a great impact on the resulting viscosity of the simulated DPD fluid. In fact, with the standard DPD method, the dissipative force vanishes, no matter how close the two particles are, whenever the relative velocity between two beads is perpendicular to their distance. To overcome this issue, Junghans and co-workers proposed an extended DPD version [21] in which the shearing dissipation is considered by evaluating the perpendicular contribution of dissipative and random forces. This extension can be interpreted as a different thermostat for DPD, a transverse DPD thermostat, that is able to describe the damping perpendicular to the interatomic axis, mimicking the shear of those degrees of freedom that were integrated out in the CGing procedure [21, 27].

A previous study tested [23] the transverse DPD thermostat applying the so-called generalized weighting function (GWF) [22, 28, 29, 30] and it was proved that it is possible to turn standard gas-like dynamics into liquid-like dynamics acting on dissipative parameters and the GWF exponent. Furthermore, it was observed that three different sets of DPD parameters lead to the simulation of a fluid with the same Sc number, radial distribution function (RDF) and effective friction coefficient[23], a measure of the overall bead friction. In particular, high Sc number fluids are particularly challenging to simulate due to the imbalanced importance of dissipative and random forces over conservative one, requiring particular attention in the investigation of the equation of motion (EOM). In the past few years, a great effort has been devoted to the design of simple, efficient and accurate numerical methods to integrate the EOM for DPD beads. All the methods exhibit the appearance of spurious effects with increasing time step size, such as the unphysical systematic drift of the temperature, uncontrolled spatial correlations among particles, errors in computed static and dynamical properties [31]. However, very small time step size significantly limits the time scales accessible for DPD simulation [32]. Nikunen et al. [33] showed that the performance of the Shardlow's scheme are superior to those of several other schemes for a number of different observables. Recently, Shardlow-like splitting algorithms have been further applied in DPD with various fixed conditions, and its parallel implementation has also been developed [34, 35, 36].

Therefore this contribution has three elements of novelty. First, the development of an automated reliable method to compute transport properties from equilibrium simulations, based on an iterative algorithm to evaluate the reliabil-

ity of the simplify strategy of the SACF. Second, the implementation in the open source software LAMMPS [37] and the performance assessment of the Shardlow-splitting algorithm formulated within the velocity-Verlet scheme (VV-SSA), in comparison with the most popular modified velocity-Verlet algorithm (VV) with application to the transverse DPD thermostat and in different friction regimes relying on systematic numerical experiments. Third, the use of the findings of the first two points to perform a multi-parametric study aiming to investigate the Sc number relationship with the effective friction coefficient.

The outline of the article is as follows. Section 2 presents the DPD EOM with the transverse DPD thermostat and the GWF, time integration schemes and the methods of calculating the transport properties. Section 3 is dedicated to computational details: the workflow of the developed algorithm for fluid transport properties calculation is illustrated and the setup of simulations is shown. In Section 4 the results are presented and discussed, leaving the conclusions in Section 5
.

2. Dissipative Particle Dynamics: A Coarse-Grained Simulation Technique

2.1. Transverse DPD thermostat: Mathematical formulation

DPD is a particle-based mesoscopic method in which groups of atoms are clustered into single particles, often called beads. The time evolution of each bead can be calculated by Newton's second law:

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{f}_i}{m_i}, \tag{2}$$

where $\mathbf{r}_i$ and $\mathbf{v}_i$ are the position and velocity of the $i-$th bead with the mass m_i, respectively. Typically in DPD, the mass of a single DPD bead, cutoff radius and thermal energy are taken as, respectively, mass, time and energy units to define DPD units. In the case of a simple DPD fluid, the force $\mathbf{f}_i$, acting on $i-$th bead is a sum of three pairwise contributions:

$$\mathbf{f}_i = \sum_{j \neq i} (\mathbf{F}^C_{ij} + \mathbf{F}^D_{ij} + \mathbf{F}^R_{ij}). \tag{3}$$

In Eq. (3), the sum runs over the indices of beads contained in the closest vicinity of bead i within a certain cutoff radius.

The conservative contribution, $\mathbf{F}^C_{ij}$, is a soft-repulsive force acting between two

beads i and j having the following functional form:

$$\mathbf{F}_{ij}^{C} = \begin{cases} a_{ij}\left(1 - \frac{r_{ij}}{r_c^C}\right)\hat{\mathbf{r}}_{ij}, & r_{ij} < r_c^C \\ 0, & r_{ij} > r_c^C \end{cases} \tag{4}$$

where a_{ij} denotes a maximum repulsion between beads i and j, $r_{ij} = |\mathbf{r}_{ij}| = |\mathbf{r}_i - \mathbf{r}_j|$ is the separation distance between a pair of the beads, $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/r_{ij}$ is the unit vector of the bead-bead separation distance and r_c^C is the cutoff radius for the conservative interactions.

Dissipative force and random forces, $\mathbf{F}_{ij}^{D}$ and $\mathbf{F}_{ij}^{R}$, respectively, represents the effects of viscosity slowing down the particles motion with respect to each other and of thermal or vibrational energy of the system acting together as a thermostat. The random and dissipative forces are coupled via the fluctuation–dissipation theorem (FDT) [25], ensuring sampling from the appropriate probability distribution while conserving total system momentum.

In this work, the extended DPD thermostat, proposed by Junghans and coworkers [38], was used. The expressions for dissipative and random forces include shear components can be written as

$$\mathbf{F}_{ij}^{D} = -\gamma_{\parallel} w_{\parallel}^2(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij} - \gamma_{\perp} w_{\perp}^2(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T)\mathbf{v}_{ij}, \tag{5}$$

$$\mathbf{F}_{ij}^{R} = \sigma_{\parallel} w_{\parallel}(r_{ij})\frac{\xi_{ij}}{\sqrt{\Delta t}}\hat{\mathbf{r}}_{ij} + \sigma_{\perp} w_{\perp}(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T)\frac{\boldsymbol{\xi}_{ij}}{\sqrt{\Delta t}}. \tag{6}$$

Here $\gamma_{\parallel}$ and $\gamma_{\perp}$ are parallel and perpendicular dissipative parameters, respectively, analogously $\sigma_{\parallel}$ and $\sigma_{\perp}$ are parallel and perpendicular noise parameters, $w_{\parallel}(r_{ij})$ and $w_{\perp}(r_{ij})$ are weighting functions for the parallel and perpendicular parts of $\mathbf{F}_{ij}^{D}$ and $\mathbf{F}_{ij}^{R}$, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$, ξ_{ij} and $\boldsymbol{\xi}_{ij}$ are scalar and vector random numbers, respectively, with zero mean value and unit variance, $\mathbf{I}$ is the identity second-rank matrix and Δt is the simulation time step.

The dissipative and noise parameters are related to each other in order to satisfy the FDT:

$$\sigma_{\alpha}^2 = 2k_B T \gamma_{\alpha}, \quad \alpha \in \{\parallel, \perp\}, \tag{7}$$

where k_B denotes the Boltzmann constant and T the system temperature. This extended DPD method results in higher fluid viscosity than the standard DPD; however, still neglecting the conservation of angular momentum. In this sense, a more complete method, called Fluid Particle Model (FPM), was proposed by Español [39], in which both shearing dissipation and conservation of angular mo-

mentum are considered. Pan and coworkers [29] proposed a modified version of FPM, which is more similar to the transverse DPD thermostat and standard DPD method, including all the modifications which have an impact on the Sc number of the investigated fluid, such as the GWF and the shearing dissipation, in a framework that preserves the total angular momentum. With this approach, the beads experience not only parallel and shear dissipation, but also a rotational damping contribution which can be properly evaluated only if the total angular momentum is conserved. The major drawback of this approach is the computational time, which is significantly higher than standard or extended DPD due to the evaluation of the rotational damping contribution. However, the requirement on angular momentum conservation is particularly important when non-equilibrium simulations are used to measure the fluid viscosity, while in this work it can be relaxed since equilibrium simulations are performed.

As previously mentioned, the GWF was introduced by Fan and coworkers [28] as:

$$w_\alpha(r_{ij}) = \left(1 - \frac{r_{ij}}{r_c^D}\right)^{s_\alpha}, \quad \alpha \in \{\parallel, \perp\}, \tag{8}$$

and its exponent, $s_\alpha \in (0,1]$, plays an important role in modeling of dynamic properties of a DPD fluid. In this work we do not study the exponents separately but we simply assume that $s_\parallel = s_\perp = s$ and consequently $w_\parallel(r_{ij}) = w_\perp(r_{ij}) = w(r_{ij})$. It should be noted that a dissipative cutoff radius, r_c^D, is introduced for dissipative and random forces in the weighting functions. This parameter has no influence on the nature of the phase assumed by the simulated system, described by the radial distribution function (RDF); conversely, it plays a role in the description of the transport properties as we will see in the following section.

2.2. Time integration schemes

As previously mentioned, the numerical integration of DPD EOM is a non-trivial task due to its stochastic nature. The pairwise coupling of particles through the random and dissipative forces requires special attention. In fact, unlike MD, in DPD there is thus no guarantee for time reversibility. In early studies of DPD, the classical Euler algorithm was used to integrate DPD EOM, suffering from an energy drift during the simulation.

The standard VV integration scheme has been shown to be relatively accurate in typical MD simulations [40]. Unlike MD, DPD includes not only the conventional conservative forces, but also dissipative and random forces, that in turn depend on the bead velocities and white noise. However, due to its simplicity

and good overall performance the VV algorithm was the starting point for the modified VV scheme [41, 31]. In an attempt to account for DPD complexity, the dissipative forces have to be updated for a second time at the end of each integration step by using the up-to-date velocities (momenta), with the first update taking place right after the positions are updated at each step as illustrated in Table 1. This improves integration performance considerably yet keeping it computationally efficient since the additional update of dissipative forces is not particularly time-consuming. For this reason, it is chosen as the reference when a comparison with other integrators is carried out.

Table 1: Outline of the modified VV integration scheme (see text for acronym) to update momentum and position of each i-particle ($\mathbf{p}_i^s \to \mathbf{p}_i^{s+1}; \mathbf{r}_i^s \to \mathbf{r}_i^{s+1}$). The scheme with detailed integration steps is available in the Supporting Information material.

1. For $i = 1, ..., N$
   a. *Momenta First Update:*
   $\mathbf{p}_i^{s+1/2} \leftarrow \mathbf{p}_i^s$; $\mathbf{f}_i^C(\mathbf{r}^s)$; $\mathbf{f}_i^D(\mathbf{r}^s, \mathbf{p}^s)$; $\mathbf{f}_i^R(\mathbf{r}^s)$,where $\mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C$, $\mathbf{f}_i^D = \sum_{j \neq i} \mathbf{F}_{ij}^D$ and $\mathbf{f}_i^R = \sum_{j \neq i} \mathbf{F}_{ij}^R$
   b. *Positions Update:*
   $\mathbf{r}_i^{s+1} \leftarrow \mathbf{r}_i^s$; $\mathbf{p}_i^{s+1/2}$
2. *Force First Calculation:* $\{\mathbf{f}_i\}_{i=1}^N$, with $\mathbf{f}_i^C(\mathbf{r}_i^{s+1})$; $\mathbf{f}_i^D(\mathbf{r}_i^{s+1}, \mathbf{p}_i^{s+1/2})$; $\mathbf{f}_i^R(\mathbf{r}_i^{s+1})$
3. For $i = 1, ..., N$
   a. *Momenta Second Update:*
   $\mathbf{p}_i^{s+1} \leftarrow \mathbf{p}_i^{s+1/2}$; $\mathbf{f}_i^C(\mathbf{r}^{s+1})$; $\mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}^{s+1/2})$; $\mathbf{f}_i^R(\mathbf{r}^{s+1})$
4. *Dissipative Force Second Calculation:* $\left\{\mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}^{s+1})\right\}_{i=1}^N$

Shardlow introduced the most recent addition to DPD integrators [34, 35, 36], applying an idea commonly used in solving differential equations to the case of integrating the EOM in DPD. The basic idea of the Shardlow-splitting algorithm (SSA) is the decomposition of the EOM [Eq. (2) and Eq. (3)] into differential equations corresponding to the deterministic dynamics and elementary stochastic differential equations (SDEs) corresponding to the fluctuation-dissipation contribution. This approach allows to adopt larger time step sizes capable of capturing mesoscale events and to recover in this way the practical computational advantage of DPD. The original SSA was formulated for systems with equal-masses particles [34] and successively extended to unequal-mass particles [35].

As reported in Table 2 the key idea is to factorize the integration process such that the conservative forces are calculated separately from the dissipative and random terms: the stochastic integration of the dissipative and random forces is per-

formed prior to the deterministic integration of the conservative force. In a typical SSA formulation, both types of differential equations are integrated via the VV algorithm [34]. However, the specific choice of the VV algorithm is not a requirement. Rather, the separated stochastic and deterministic dynamics components may be solved using any traditional MD numerical integration scheme. In this work, the stochastic and deterministic differential equation are integrated with the SSA formulated for the VV algorithm (VV-SSA).

First, based upon time-splitting, the SSA decomposes the EOM into: differential equations that correspond to the deterministic dynamics; and SDEs that correspond to the stochastic dynamics. Second, based upon operator splitting, the particle stochastic dynamics are updated in a pairwise manner that conserves linear momentum [35, 36].

Table 2: Outline of the VV-SSA integration scheme to update momentum and position of each i-particle ($\mathbf{p}_i^s \rightarrow \mathbf{p}_i^{s+1}; \rightarrow \mathbf{r}_i^s \rightarrow \mathbf{r}_i^{s+1}$) (see text for acronym). The scheme with detailed integration steps for the transverse DPD thermostat is available in Supporting Information material.

1. *Stochastic Integration: For all $i-j$ pairs of particles*
   a. $\mathbf{p}_i^{s+1/4} \leftarrow \mathbf{p}_i^s; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^s); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
   b. $\mathbf{p}_j^{s+1/4} \leftarrow \mathbf{p}_j^s; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^s); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
   c. $\mathbf{v}_{ij}^{s+1/4} \leftarrow \frac{\mathbf{p}_i^{s+1/4}}{m_i} - \frac{\mathbf{p}_j^{s+1/4}}{m_j}$
   d. $\mathbf{p}_i^{s+2/4} \leftarrow \mathbf{p}_i^{s+1/4}; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^{s+1/4}); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
   e. $\mathbf{p}_j^{s+2/4} \leftarrow \mathbf{p}_j^{s+1/4}; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^{s+1/4}); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
2. *Deterministic integration* #1: For $i = 1, ..., N$
   a. $\mathbf{p}_i^{s+3/4} \leftarrow \mathbf{p}_i^{s+2/4} + \frac{\Delta t}{2}\mathbf{f}_i^C(\mathbf{r}^s)$
   b. $\mathbf{r}_i^{s+1} \leftarrow \mathbf{r}_i^s + \Delta t \frac{\mathbf{p}_i^{s+3/4}}{m_i}$
3. *Conservative Force Calculation:* $\left\{\mathbf{f}_i^C(\mathbf{r}^{s+1})\right\}_{i=1}^N$, where $\mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C$
4. *Deterministic integration* #2: For $i = 1, ..., N$
   $\mathbf{p}_i^{s+1} \leftarrow \mathbf{p}_i^{s+3/4} + \frac{\Delta t}{2}\mathbf{f}_i^C(\mathbf{r}^{s+1})$

2.3. Evaluation of transport coefficients in DPD simulations

Fluid transport properties are some of the numerous properties that can be evaluated from molecular simulations, either standard at micro-scale level or at meso-scale with any CGing approach. The estimation of transport coefficients in DPD simulations takes place with numerical techniques developed for atomistic simulations. The viscosity can be calculated from equilibrium and non-equilibrium simulations. In the first case, all quantities are measured when the

system has reached its thermodynamic equilibrium; in the latter, the particles are constantly perturbed by an external field applied to the system. In the case of equilibrium simulations, the obtained value is the so-called zero-shear viscosity and it can obtained using the GK relations. The numerical integration of the SACF results in zero-shear viscosity. Other properties related to the average distance travelled by the particles can be extracted analogously. For instance, the diffusion coefficient can be calculated by the integral of the velocity autocorrelation function (VAF) according to the GK approach.

The self-diffusion coefficient in this work is calculated by the Einstein relation [42]. In this method, the self-diffusion coefficient, proportional to the mean-square displacement (MSD) of the beads and for the bulk phase, is expressed as follows:

$$\mathscr{D} = \lim_{t \to +\infty} \frac{< [\mathbf{r}(t_0 + t) - \mathbf{r}(t_0)]^2 >}{6t}, \tag{9}$$

where $< \cdots >$ denotes an ensemble average.
GK [43, 44] relation allows to calculate the viscosity by integrating the SACF as

$$\eta = \frac{V}{k_B T} \int_0^{+\infty} < \sigma_{\alpha\beta}(t_0) \sigma_{\alpha\beta}(t_0 + t) > dt, \quad (\alpha \neq \beta), \tag{10}$$

where V is the system volume and $\sigma_{\alpha\beta}$ is the off-diagonal component of the stress tensor.

It is worth mentioning that alternative GK formulas exist [38, 45]:

$$\eta = \eta_\infty + \frac{V}{k_B T} \int_0^{\infty} \left\langle \left(\sigma^C_{\alpha\beta}(t_0 + t) + \sigma^D_{\alpha\beta}(t_0 + t) \right) \left(\sigma^C_{\alpha\beta}(t_0) - \sigma^D_{\alpha\beta}(t_0) \right) \right\rangle dt \tag{11}$$

which is equivalent to

$$\eta = \eta_\infty + \frac{V}{k_B T} \int_0^{\infty} \left\langle \left(\sigma_{\alpha\beta}(t_0 + t) - \sigma^R_{\alpha\beta}(t_0 + t) \right) \sigma_{\alpha\beta}(t_0) \right\rangle dt \tag{12}$$

They were introduced by splitting the stress tensor into conservative, dissipative, and random force contributions as

$$\begin{aligned} \sigma_{\alpha\beta} &= \frac{1}{V} \sum_i \left(m_i v_{i,\alpha} v_{i,\beta} + \sum_{j<i} r_{ij,\alpha} F^C_{ij,\beta} \right) + \frac{1}{V} \sum_i \sum_{j<i} r_{ij,\alpha} F^D_{ij,\beta} + \frac{1}{V} \sum_i \sum_{j<i} r_{ij,\alpha} F^R_{ij,\beta} \\ &= \sigma^C_{\alpha\beta} + \sigma^D_{\alpha\beta} + \sigma^R_{\alpha\beta} \end{aligned} \tag{13}$$

and pre-averaging correlations due to the random contribution as

$$\eta_\infty = \frac{V}{k_B T}\left\langle \left(\sigma^R_{\alpha\beta}\right)^2 \right\rangle \frac{\delta t}{2} \tag{14}$$

Alternatively, the Einstein-Helfand (EH) formula

$$\eta = \frac{V}{k_B T}\lim_{t\to\infty}\frac{1}{2t}\left\langle \left(\int_0^t \sigma_{\alpha\beta}(t')dt'\right)\left(\int_0^t \sigma_{\alpha\beta}(t')dt'\right)\right\rangle \tag{15}$$

can also be straightforwardly and safely used with a high degree of accuracy to compute η in DPD [46]. Its use does not require any decomposition of the stress tensor to reduce the uncertainty due to the noise or any modification of the SACF. Although the alternative GK formulas (11) or (12) and/or the EH formula (15) be theoretically superior to the traditional GK formula (9), we employed traditional GK formula, Eq. (9), since it allows on-fly evaluation of the SACF necessary in our automated procedure without necessity of an additional post-processing.

In DPD simulations, the stress tensor can be evaluated by the Irving-Kirkwood expression:

$$\sigma_{\alpha\beta}(t) = \frac{1}{V}\sum_i \sum_{j<i} r_{ij,\alpha}(t) F_{ij,\beta}(t) + \frac{1}{V}\sum_i m v_{i,\alpha}(t) v_{i,\beta}(t), \tag{16}$$

where $F_{ij,\beta} = F^C_{ij,\beta} + F^D_{ij,\beta} + F^R_{ij,\beta}$, and subscripts α and β are reserved for vector or tensor indices while subscripts i and j for bead indices. For practical calculations, the evaluation of the SACF is performed not using its definition, which requires to store in memory the velocity of all particles at every time step, but an approximate and cheaper algorithm. The one adopted in the open source atomistic code LAMMPS [37] makes use only of a limited data set decided by the user to carry out the SACF evaluation. Indeed, the SACF averages are computed every $\mathrm{N_{freq}}$ time steps and only a limited data set is actually employed in the SACF evaluation, namely the data points correlated every $\mathrm{N_{ev}}$-th time step for $\mathrm{N_{ev}} \times (\mathrm{N_{rep}} - 1)$ time steps. The resulting correlation data, $C_{\alpha\beta}$, is calculated as follows:

$$C_{\alpha\beta}(t) = \left\langle \sum_{\alpha\neq\beta} \sigma_{\alpha\beta}(t)\sigma_{\alpha\beta}(t+\delta) \right\rangle. \tag{17}$$

The ensemble average in Eq. (17) is calculated on each pair of input values separated by the time delta δ. The maximum value of δ used is the instant of time

equal to $N_{ev} \times (N_{rep} - 1)$ time integration steps. Therefore, the amount of correlation data calculated is equal to $N_{rep} : C_{\alpha\beta}(0),\ C_{\alpha\beta}(N_{ev}),\ C_{\alpha\beta}(2 \times Nev), \ldots, C_{\alpha\beta}((N_{rep} - 1) \times N_{ev})$. The correlation data are calculated for each $i = 0, 1, 2, \ldots, (N_{rep} - 1)$ using the following expression:

$$C_{\alpha\beta}(N_{ev} \times i) = \frac{1}{k+1} \sum_{j=0}^{k} \sigma_{\alpha\beta}(N_{ev} \times j) \times \sigma_{\alpha\beta}(N_{ev} \times j + N_{ev} \times i) \,, \tag{18}$$

where $k = (N_{freq}/N_{ev}) - i$. For further details on this algorithm readers are referred to the LAMMPS code and documentation [37]. This approximate method therefore introduces three parameters N_{freq}, N_{ev} and N_{rep}, which values greatly affect both accuracy and computational time of the SACF computation. Moreover, these values are system-dependent, since a high-correlated fluid requires to use a low value of N_{ev} and high values of N_{freq} and N_{rep} to properly capture the decorrelation dynamics. This motivated us in the development of an automated strategy to find these parameters for a generic system, as the reader may see in the following.

2.4. Development of an automated method for transport properties calculation

In some research works [45, 19, 20], the reliability of GK relations in DPD simulations was doubted, since the presence of stochastic forces creates noise to the resulting stress tensor, generating high statistical errors in the viscosity calculation from the SACF. To make clearer the rationale behind the method developed in this work, it is worth quickly remarking the main issues related to the GK method in DPD simulations.

The focal points for the viscosity calculation based on GK method are the achievement of a good statistics in the SACF with the minimum computational cost, and the proper treatment of the noise in the numerical evaluation of the time integral of the SACF. The calculation of such time integral with a simple trapezoidal rule is affected by the SACF tail oscillations and therefore by the selection of the last integration point. To overcome these issues, in our previous work [23] a procedure based on the calculation of the Cumulative Viscosity Integral in function of time (CVI) was proposed and this work aims to automatize and optimize it. Particular attention was devoted to this aspect since the collective correlations, such as the SACF, in literature studies are often determined with poorer statistics than single-particle ones, such as the velocity autocorrelation function, because of the extensive computational requirements. As previously mentioned, the proper

evaluation of the SACF, and in turn of the viscosity, using the approximate algorithm implemented in the open source code LAMMPS depends on the choice of the parameters N_{ev}, N_{rep} and N_{freq}. Since more data should be accumulated to improve the statistics, thus small accumulation frequencies are preferred and in this work $N_{ev} = 1$. Furthermore, N_{freq} is always taken equal to N_{rep}. Also the simulation length affects the statistics, and is given by $N_{sim} \times N_{freq}$, where N_{sim} is the number of times the SACF averages are computed.
As it will be shown in the following, for each DPD system there is a certain minimum N_{rep} value that guarantee SACF values approach to zero in the decorrelation time interval, and a certain minimum N_{sim} value sufficiently high to smooth instantaneous tail fluctuations giving an accurate solution. The identification of these parameters is not a trivial-task, and a trial-and-error approach can lead to a waste of computational resources. The newly developed method overcomes this problem, since it is based on an algorithm that calculates automatically the minimum values of N_{rep} and N_{sim} in a DPD simulation of a fluid in order to deliver a value of viscosity within an acceptable range of uncertainty. The algorithm also identifies the simulation parameters required to produce the shortest trajectory to achieve the fixed statistical accuracy in the calculation of the SACF and, consequently, of the viscosity.

The automated method makes use of LAMMPS for performing DPD simulations and of a Python script for the management of simulations and post-processing. The workflow of the method is schematized in Fig. 1, while its pseudo-code is reported in Algorithm 1. The algorithm is based on three main iterative loops: an external control loop and two internal consecutive control loops. The first is indicated as the Δt-loop and the internal loops as N_{rep}-loop and N_{sim}-loop. The Δt-loop executes different simulations progressively reducing the time step size, in order to find the largest time step for which the viscosity value does not change anymore, namely a time step independent value of viscosity η_{fin}, which is the objective of the aforementioned external loop and more in general of the automated algorithm.

The two inner control loops instead, the N_{rep}-loop and the N_{sim}-loop, give at the end a value of viscosity, called intermediate viscosity, η_{int}, which is the out of loop condition when both subsequent loops are satisfied. The first control loop, the N_{rep}-loop, progressively increases the value of N_{rep} starting from an arbitrary initial guess of N_{rep}, N_{sim} and Δt, and checks if CVI has the expected asymptotic behavior, with the final points of the curve that do not change anymore with time. This is equivalent to capture entirely the expected behavior of the SACF, from the initial condition of maximum stress correlation up to complete decorrelation. If

the time interval analyzed by the SACF is not sufficiently large to trace the whole decorrelation dynamics, the tail of the SACF may not reach zero, in turn leading to a CVI curve that still increases in function of time. The size of the time interval to be analyzed is determined by the choice of N_{rep}, which is the value that the N_{rep}-loop tries to identify. The optimal N_{rep} value is the one that includes the full decorrelation behaviour of the SACF in the shortest analyzed time interval. To establish if the CVI asymptote is reached, a condition referred in the following as viscosity plateau, the algorithm uses a special adaptation of the Simple Moving Average method. This method creates, for each value of N_{rep}, a series of averages, η_{avg} of different time intervals (*Averaging Windows* or *Subsets*) of a fixed extension (*Window Size*) of the full CVI curve and checks the relative error, RE_{CVI}, between them. Specifically, the user must define the distance between subsets, i.e., the *Sliding Window*, that is moved forward in the calculation of the *Current Subset* after the calculation of the average on the previous one, *Reference Subset*. A certain number of subsets, defined by the user through the *Jump Index*, is jumped between the *Reference Subset*=*Current Subset* - *Jump Index* and *Current Subset*, to avoid the possible but undesired identification of a local plateau in the CVI. If the value of the RE_{CVI} is over a chosen threshold, another simulation with an increased value of N_{rep} with increment h_{rep} is performed. This iterative mechanism continues until the RE_{CVI} is below the desired threshold: when this eventually happens the algorithm passes to the next control loop the viscosity plateau, η_{plt} and the reference of its subset, indicated as *Plateau Subset*, together with the associated standard deviation σ_{plt}. In this way, the point up to which integrate the SACF is automatically established and indicated as the viscosity UI point, the upper integration bound of the viscosity integral. There exists the possibility that η_{plt} is not reached in a maximum number of iterations, again decided by the user: in this case, the algorithm firstly increments N_{sim} and then restarts the N_{rep}-loop from the last N_{rep} investigated. We encountered this situation only in a few times during our extensive testing of the algorithm, mainly with fluids with very high viscosity, because such fluids requires from the beginning a higher guess value for N_{sim}. The N_{sim}-loop has a similar structure of the N_{rep}-loop but it serves a different purpose. The act of increasing the N_{sim} value of h_{sim} is performed in order to reduce the noise of the CVI, quantified with the coefficient of variation (COV) of the η_{plt}, namely the ratio between the σ_{plt} and the η_{plt}. The check is done on the absolute error between the COV of two consecutive simulations, and this quantity must be lower than a certain threshold defined by the user, AE_{COV}. Eventually the output of the N_{sim} loop is the previously mentioned η_{int} associated to the current Δt, which is than fed to the external Δt-loop. The Δt-loop keeps iterating until the

relative error between two consecutive η_{int} values reaches a threshold specified by the user, $\text{RE}_{\eta_{\text{int}}}$.

The algorithm returns a value of η_{fin} sufficiently accurate, according to the value of $\text{RE}_{\eta_{\text{avg}}}$, AE_{COV} and $\text{RE}_{\eta_{\text{int}}}$ that the user has decided to set. The choice of the N_{rep}, N_{sim} and Δt initial guest is arbitrary done by the user and the results do not depend on their choice. However, the time-to-solution of the algorithm depends on the goodness of the initial guest setup that should be guided from preliminary tests on the system of interest, as done in this work. The values of h_{rep} and h_{sim} have to be significant for the system simulated. In this work $\text{h}_{\text{rep}} = 500$ and $\text{h}_{\text{sim}} = 50$ are chosen as increments of the respective loops, since resulting, from a preliminary study on the SACF behaviour in the case of simple fluids, as the minimum values giving a significant improvement of the SACF statistic. The threshold values for the out-of-loop conditions $\text{RE}_{\eta_{\text{avg}}}$, $\text{RE}_{\eta_{\text{int}}}$ and AE_{COV} are chosen as a compromise between the desired accuracy of the viscosity estimate and the computational cost required. Indeed, the checks performed by the automated method on the computed viscosity through the three out-of-loop conditions fix the level of acceptability of the error associated resulting from the whole numerical methodology. The error in the GK approach is mainly due inaccuracies in the calculation of the SACF. Such error is quantified with the coefficient of variation (COV) of the viscosity estimate, namely the ratio between the standard deviation and the mean viscosity value. The calculation of the viscosity is reiterated with different combination of the parameters defining the time interval analyzed by the SACF and the simulation time, until the condition on its coefficient of variation is satisfied. The same procedure is repeated with different time step sizes until the time resolution suitable for the specific system simulated is reached. The optimum threshold values should be the highest below which there is no significant error reduction. In this work we found that $\text{RE}_{\eta_{\text{avg}}} = 0.01$, $\text{RE}_{\eta_{\text{int}}} = 0.05$ and $\text{AE}_{\text{COV}} = 0.01$ work best in our investigation of a simple DPD fluid with Schmidt number from about 1 to 3881.

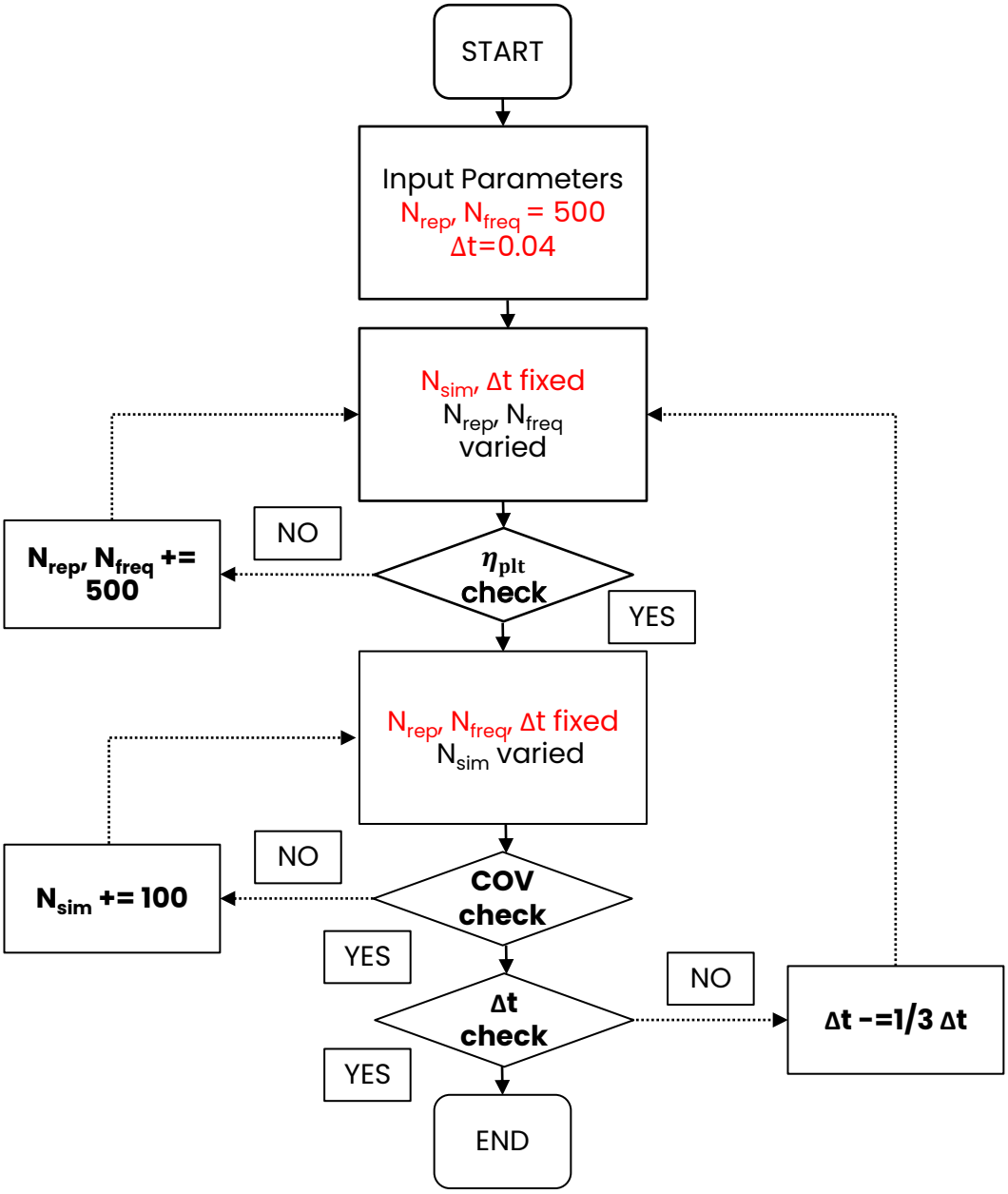

Figure 1: Workflow of the automated method for viscosity calculation from DPD simulation with the Green-Kubo approach.

Algorithm 1 Pseudo-code of the algorithm for automated viscosity calculation from DPD simulation with the Green-Kubo approach.

Setup of the DPD system specific input parameters a,s,γ,r_c^D
Setup of the initial guest for N_{rep}, N_{sim} *and* Δt *to initiate the* 1^{st} *simulation*

\# *Starting the* Δt-loop
while True **do**

  \# *Starting the* N_{rep}-loop
  while True **do**
    • Run simulation with the *Current* N_{rep}
    • Calculation of the CVI from the SACF (at the *Current* N_{rep})
    • Adaptation of the Simple Moving Average method to search η_{plt}
    • Calculation of different η_{avg} and σ_{avg} for different (*Averaging Windows*)

    for each η_{avg}(*Current Subset*) **do**
      if $\frac{|\eta_{avg}(Current\ Subset) - \eta_{avg}(Current\ Subset - Jump\ Index)|}{\eta_{avg}(Current\ Subset - Jump\ Index)} <= RE_{\eta_{avg}}$ **then** ▷ Out-of-N_{rep}-loop condition

        break
        ✓N_{rep} identified
        ✓*Plateau Subset* ← *Current Subset*
        ✓$\eta_{plt} \leftarrow \eta_{avg}$
        ✓$\sigma_{plt} \leftarrow \sigma_{avg}$

      end if

    end for

    • $N_{rep} \leftarrow N_{rep} + h_{rep}$

  end while
  \# *End of the* N_{rep}*-loop* with *Output:* N_{rep}, *Plateau Subset*, η_{plt}, σ_{plt}

  \# *Starting the* N_{sim}-loop
  while True **do**
    • Run simulation with the *Current* N_{sim}
    • Calculation of the CVI from the SACF (at the *Current* N_{sim})
    • Calculation of η_{plt} and σ_{plt} on the *Plateau Subset*
    • Calculation of the $COV_{plt} = \frac{\sigma_{plt}}{\eta_{plt}}$
    if $|COV(Current\ Simulation) - COV(Previous\ Simulation)| < AE_{COV}$ **then**
    ▷ Out-of-N_{sim}-loop condition

      break
      ✓N_{sim} identified
      ✓$\eta_{int} \leftarrow \eta_{plt}$
      ✓$COV_{int} \leftarrow COV_{plt}$

    end if

    • $N_{sim} \leftarrow N_{sim} + h_{sim}$

  end while
  \# *End of the* N_{sim}*-loop with Output:* N_{sim}, η_{int}, COV_{int}

  if $\frac{|\eta_{int}(Current\ Simulation) - \eta_{int}(Previous\ Simulation)|}{\eta_{int}(Previous\ Simulation)} <= RE_{\eta_{int}}$ **then** ▷ Out-of-Δt-loop condition

    break
    ✓Δt identified
    ✓$\eta_{fin} \leftarrow \eta_{int}$

  end if

  • $\Delta t \leftarrow \Delta t - \frac{1}{3}\Delta t$

end while
\# *End of the* Δt*-loop* with *Output:* Δt, η_{fin}

3. Computational details

3.1. Simulation setup details

All simulations were performed by using the open-source software LAMMPS [37]. DPD simulations are setup by means of a script file which contains all the input parameters and the instructions for the calculations. The developed algorithm bases the entire procedure of management and submission of simulations by means of automatic generation of these files. In this section simulation features kept constant in all simulations are reported. A system of $N = 10125$ identical particles ($m_i = m = 1$) was simulated in a cubic box of the length equal to 15 DPD length units with periodic boundary in all directions, considering r_c^C as the DPD length unit. In all simulations the potential repulsion parameter $a_{ii} = a$ was set equal to 25 [7]. DPD fluid was represented by single beads with degree of CGing, $N_m = 1$, which affect the evaluation of the repulsion parameter a of the conservative force, i.e.,

$$a = \frac{N_m \kappa^{-1} - 1}{0.2\rho} k_B T, \tag{19}$$

where N_m corresponds to the number of real molecules per bead, κ^{-1} is the inverse of the dimensionless isothermal compressibility of DPD fluid (e.g., $\kappa^{-1} = 16$ for water) and ρ is the DPD number density corresponding to the number of beads inside a volume of $(r_c^C)^3$. The system bead density and the system temperature were the same for all the cases and set to $\rho = 3$ and $k_B T = 1$.

3.2. Integration algorithms: performance comparison

In this study some numerical experiments were performed to test the time step size stability threshold and the performance of the VV-SSA algorithm, newly implemented in LAMMPS by us for the `dpdext/fdt pair_style` in comparison with the most popular VV integrator used in many previous papers. Several studies evaluated the integrators, as traditionally done, based on temperature conservation [32, 33, 47]. Stability and accuracy can be achieved using longer time steps with the VV-SSA [36]. As the friction increases in the system the VV algorithm takes more iterations to stabilize the temperature [48] with a smaller time step compared to low friction condition [31, 41]. Although evaluation against dynamic fluid transport properties is essential to have a measure of the dynamical fidelity that numerical methods are able to reproduce, few such studies limited to low friction regime were performed. Furthermore the comparison between VV-SSA and other integration schemes was presented only with application to the

standard DPD thermostat [32, 48]. Since to have a deeper understanding of how the physical system evolves it is not enough to just evaluate static quantities, in this work particular attention was devoted to the performance of the investigated time integration schemes in fluid transport properties evaluation with application to the transverse DPD thermostat. In particular, it was examined the behaviour of the VV-SSA in comparison to the VV for measuring the SACF, which is related to the zero shear viscosity η. Of course, also important static quantities, such as radial distribution function where considered in the analysis. The comparison between integrators was performed investigating also the effect of the dissipative strength γ, the exponent of the generalized weight function s and the dissipative cutoff radius r_c^D.

3.3. Parametric study

Once a reliable automatized procedure for calculating the viscosity was established, a high number of simulations to test extensively the newly implemented code was performed. The effect of different DPD parameters on the Sc number of a simple DPD fluid was investigated. The considered parameters are the dissipative cutoff radius, r_c^D, the weighting function exponent, s and the dissipative parameters, $\gamma_{\parallel}$ and $\gamma_{\perp}$. For simplicity, it was considered $\gamma_{\perp} = \gamma_{\parallel} = \gamma$. The effects of s, γ, and r_c^D can be grouped into one single characteristic quantity measuring the overall bead friction, the so-called "effective friction coefficient", defined as follow:

$$\gamma_{\text{eff}} = 4\pi \int_0^{r_c^D} \gamma w(r_{ij})\, g(r_{ij}) r^2 dr_{ij}, \tag{20}$$

where $g(r_{ij})$ is the RDF. The simulations cover a large range of Sc from $O(10)$ to $O(10^3)$. A summary of all DPD simulations performed is provided in the Supporting Information. As seen below, the SSA-VV performed better in the larger friction region, allowing the exploitation of the DPD parameter space, inaccessible in previous studies [23] due to the limitations emerged using the VV integrator.

4. Results and discussion

4.1. The effect of time integration schemes on SACF calculation

Fig. 2 compares the performance of the two time integration schemes investigated, the VV and the VV-SSA, in terms of SACF calculation. This analysis has been carried out by considering three different simple fluids, having three different values of γ_{eff}, resulting in three different Sc numbers. In this preliminary study,

the values of N_{rep} and N_{sim} used to numerically evaluate the SACF are sufficiently high to ensure a significant statistics. For very small values of the effective friction coefficient, Fig. 2 shows almost identical decay profiles. While the SACF is significantly different in the high friction region. The performance difference between the two integrators becomes more evident as the γ_{eff} of the system increases, requiring the use of smaller time step sizes. The higher friction results in a higher dissipation and hence in a faster decorrelation of shear stresses in the system, requiring a smaller time step size in order to have a better temporal resolution of the SACF. Indeed, the smallest relevant time scale in the system is the momentum relaxation time, τ_m which is related to the friction in the system, and can be evaluated as follows [49]:

$$\tau_m = m\gamma^{-1}\rho^{-1}r_c^{-3}. \tag{21}$$

Comparing the definitions of the γ_{eff} and τ_m in Eqs. (20) and (21), it is evident that τ_m is proportional to γ_{eff}^{-1}. The time step sizes required for the three values of γ_{eff} investigated are reported in Table 3 and the expected trend is recovered. Furthermore, since the γ_{eff} is a measure of the overall bead friction in the system, the Sc is expected to increase for higher values of the γ_{eff}. The Sc numbers obtained for the three investigated values of γ_{eff} are in line with the expectation. One can conclude that the simulation of high Schmidt number fluids requires the use of smaller time step size compared to low Schmidt number fluids. The VV displays integrator-induced artifacts as the γ_{eff} and the Sc increases requiring smaller time step sizes. The SACF was found to be subject of increasing fluctuations with increasing friction in the case of the VV integrator. The mean of these fluctuations was close to zero at long times. Furthermore, as friction increases, the VV integrator shows increased zero-time stress autocorrelation values without a predictable trend ($SACF_0$) as reported in Table 3, while the value for the VV-SSA remains the same as evident in Fig. 3.

Table 3: Effect of the γ_{eff} on the zero-time stress autocorrelation value for VV and VV-SSA integrators.

γ_{eff}	Sc(VV-SSA)	Δt	$SACF_0$(VV)	$SACF_0$(VV-SSA)
1.092	1.103	0.0119	0.0352	0.0019
29.38	86.368	0.0079	3.2113	0.0019
58.76	336.425	0.0053	14.4522	0.0019

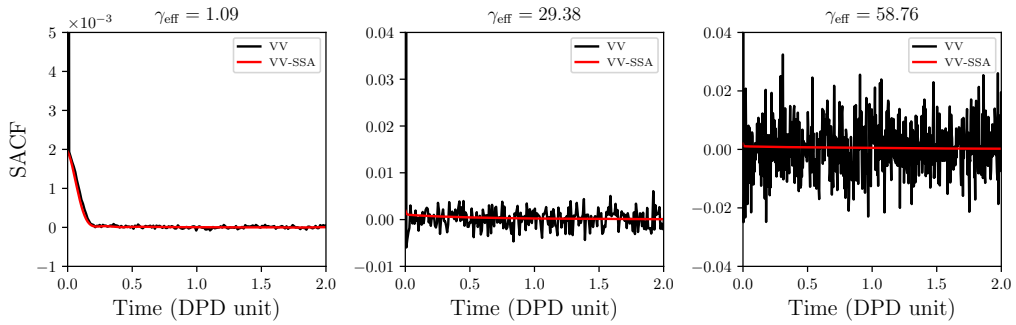

Figure 2: Comparison of SACF for VV and VV-SSA integrators at different values of γ_{eff}; $\gamma_{\text{eff}} = 1.09$ (left), $\gamma_{\text{eff}} = 29.38$ (center) and $\gamma_{\text{eff}} = 58.76$ (right).

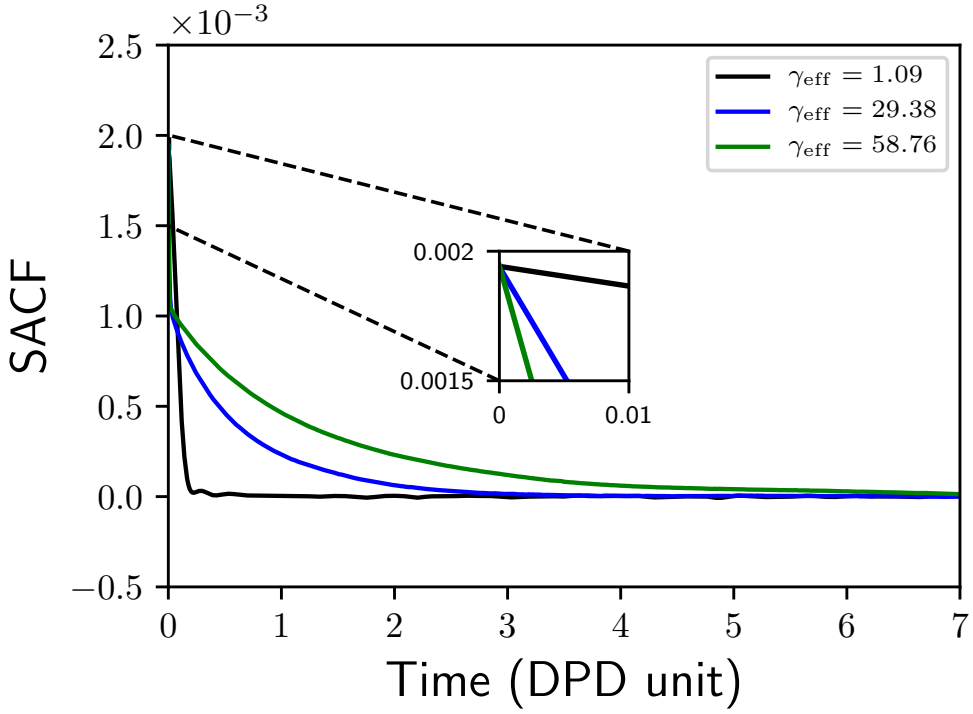

Figure 3: Comparison of SACFs measured with VV-SSA at different values of γ_{eff}.

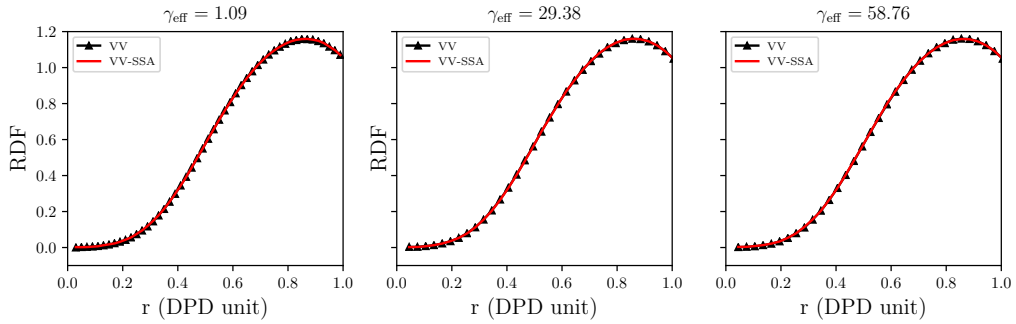

Figure 4: Comparison of RDF for VV and VV-SSA integrators at different values of γ_{eff}; γ_{eff} = 1.09 (left), γ_{eff} = 29.38 (center) and γ_{eff} = 58.76 (right).

Fig. 4 compares the radial distribution function (RDF) that characterizes the structure of the fluids. The two integrators display exactly the same RDF for each effective friction coefficient value investigated, indicating that different systems have the same structure, as expected, despite of the employed integrator and the Sc. This is expected since the three investigated cases have the same conservative force parameters.

Fig. 5 displays the effect of the time step size for a fixed value of the γ_{eff}, namely γ_{eff} = 29.38, on the SACF for the VV and the VV-SSA integrators. The frequency of SACF data accumulation and the time step size of integration affect the temporal resolution of the SACF. In this work N_{ev} is fixed to 1 as better explained in Section 2 and the effect of the time step size is investigated. In the case of the VV, the time step size affects the zero-time stress autocorrelation value and the noise amplitude in the stress tensor. In particular, for smaller time step sizes the zero-time stress autocorrelation value increases as reported in Table 4 and larger fluctuations appear because of the better temporal resolution. In the case of the VV-SSA, the time step size has no effect on the zero-time SACF value. Instead, the subtended area increases as the time step size decreases until a time step value is reached such that the curve does not change anymore. As it can be observed from the results, the VV-SSA is able to compute the SACF with the proper temporal resolution without the appearance of integrator-induced artifact, with clear advantages in the calculation of viscosity from the SACF.

4.2. Automated transport properties calculation

Following the conclusion of Section 4.1, all the results showed in the current and in the next paragraphs were produced using the VV-SSA as time integrator.

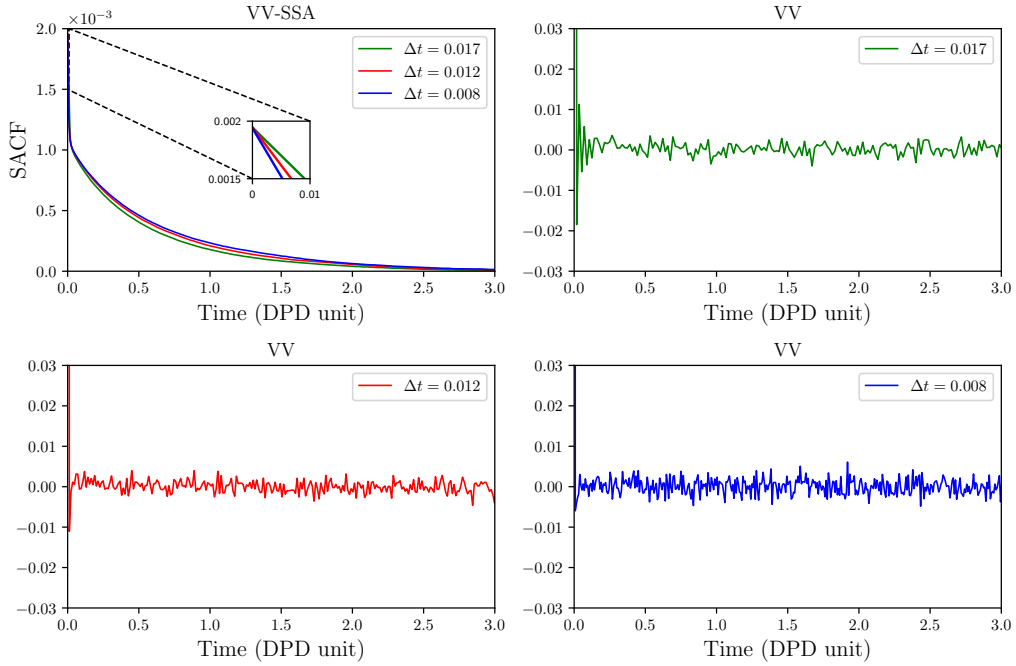

Figure 5: Time step size effect on SACF for VV and VV-SSA integrators at $\gamma_{\text{eff}} = 29.38$. Inset: independency of SACF initial point from time step size for the VV-SSA integrator.

Table 4: Effect of the time step size on the zero-time stress autocorrelation values (SACF_0) at $\gamma_{\text{eff}} = 29.38$

Δt	SACF_0(VV)	SACF_0(VV-SSA)
0.017	1.438	0.0019
0.012	2.147	0.0019
0.008	3.211	0.0019

This paragraph presents the intermediate results obtained for a particular DPD system with $\gamma_{\text{eff}} = 89.254$, with the aim of showing how the automated method to calculate the viscosity works, highlighting the choices made in the three loops described in Section 2.4.

For each Δt, the sampling interval for the computation of SACF, i.e., $N_{\text{rep}} \times \Delta t$, has to be the minimum needed to capture the entire trend of the SACF, guaranteeing the achievement of the viscosity plateau by means of the N_{rep}-loop. The results obtained during a N_{rep}-loop in terms of the SACF and CVI are reported in Fig. 6. In particular, Fig. 6 displays both the SACF curves produced for different values of N_{rep} and the correspondent CVI curves. The indicators in the SACFs are the viscosity UI points for different values of N_{rep}. As it can be seen, the last iteration of the N_{rep}-loop is $N_{\text{rep}} = 3000$, for which the condition on RE_{CVI} imposed by the user is satisfied, and the SACF mean values approach to zero. In this way, the N_{rep}-loop automatically identifies as final viscosity UI point the blue one, ensuring the convergence of the CVI to a constant value. The average value of the SACF tail will be zero but it will be affected by instantaneous fluctuations that can be smoothed extending the simulation length given by $N_{\text{sim}} \times N_{\text{freq}}$.

Fig. 7 shows the SACF and CVI curves produced progressively increasing the number of iterations N_{sim}, highlighting the effect of the tail oscillation reduction on the computed values of viscosity. As it can be seen, the number of iterations performed in the N_{sim}-loop greatly affects the shape of the SACF and CVI, leading to different values of viscosity. The N_{sim}-loop stops the iterations when both the condition on RE_{CVI} and on AE_{COV} are sequentially satisfied. In this case this condition is reached when $N_{\text{sim}} = 700$, since the curves at $N_{\text{sim}} = 650$ and 700 show little differences. Fig. 8 reports that for the last value of N_{sim} both mean value of viscosity and the COV reach a plateau, where the blue lines indicate the values of η_{fin} and COV_{fin} identified by the algorithm. The computed viscosity, $\eta_{\text{fin}} = 6.694 \pm 0.002$, has a very low uncertainty resulting from the whole numerical methodology. In fact, although the higher friction regime, the value of the associated standard deviation, $\sigma_{\text{fin}} = 0.2\%$, is lower or comparable to those obtained in our previous work [23], which exhibits excellent agreement between the Green-Kubo approach and more accurate standard techniques used as benchmark.

Therefore, for each Δt the SACF is generated with the shortest trajectory giving the fixed statistical accuracy quantified in the COV associated to calculated the viscosity value by means of the N_{rep}-loop and the N_{sim}-loop.
These *optimized* curves still have a time step size dependency. It is required a certain level of temporal resolution different for each system studied, and difficult to predict a priori. Therefore, the method search for the maximum time step size

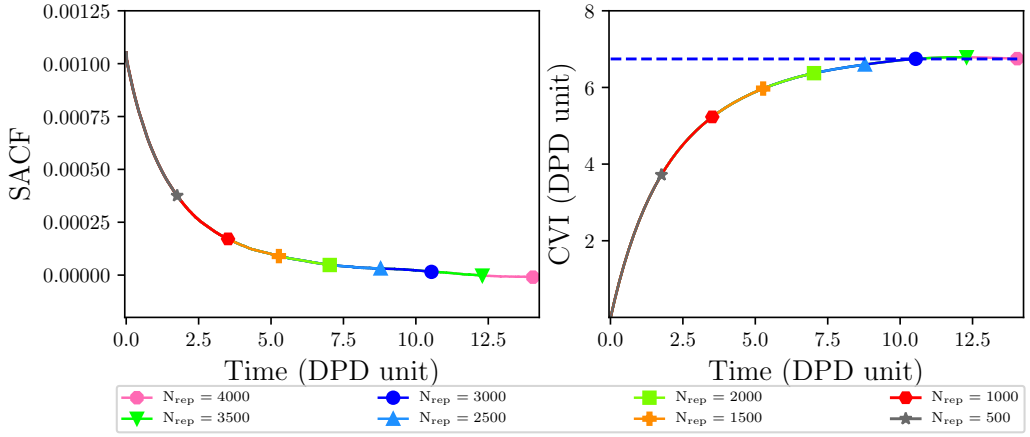

Figure 6: N_{rep}-loop: SACF curves produced for different values of N_{rep} (left) and the correspondent CVI curves (right). The indicators in the SACFs are the viscosity UI points for different values of N_{rep}. The N_{rep}-loop stops at $N_{rep} = 3000$ and identifies as UI point the blue one.

to reach this scale. Fig. 9 shows the SACF curves produced for the different time step sizes investigated by the algorithm through the Δt-loop and the correspondent values of η_{int} computed. The Δt-loop stops at $\Delta t = 0.003$, when $RE_{\eta_{int}}$ achieves the value fixed by the user, and returns the value of η_{fin} which is the blue one.

The algorithm returns for each DPD system studied the value of the viscosity η with the associated COV, $\mathscr{D}$ and Sc. Furthermore, the algorithm returns the values of Δt, N_{rep} and N_{sim} able to give time step size independent fluid transport properties. Such data are reported in the Supporting Information material of the article for all the systems investigated in this work, with the γ_{eff} ranging from 1 to 220.

To ensure the correctness of the automated algorithm, it is essential to check that the identified time step size gives meaningful results also in terms of static properties. In particular, the system temperature should be equal to the target, with an acceptable deviation of 2% according to the criteria proposed by Groot and Warren [7]. The kinetic temperature control of the VV-SSA scheme for each time step size investigated is shown in Table 5. As it can be seen, the temperature fluctuations were lower than about 2% of k_BT.

Fig. 10 compares the radial distribution function obtained with different time step sizes investigated by the algorithm during the $\Delta t-$loop. All the curves are overlapping, which indicates that the impacts of Δt on the structure of the fluid is negligible. This means also that the investigated time steps are already quite low,

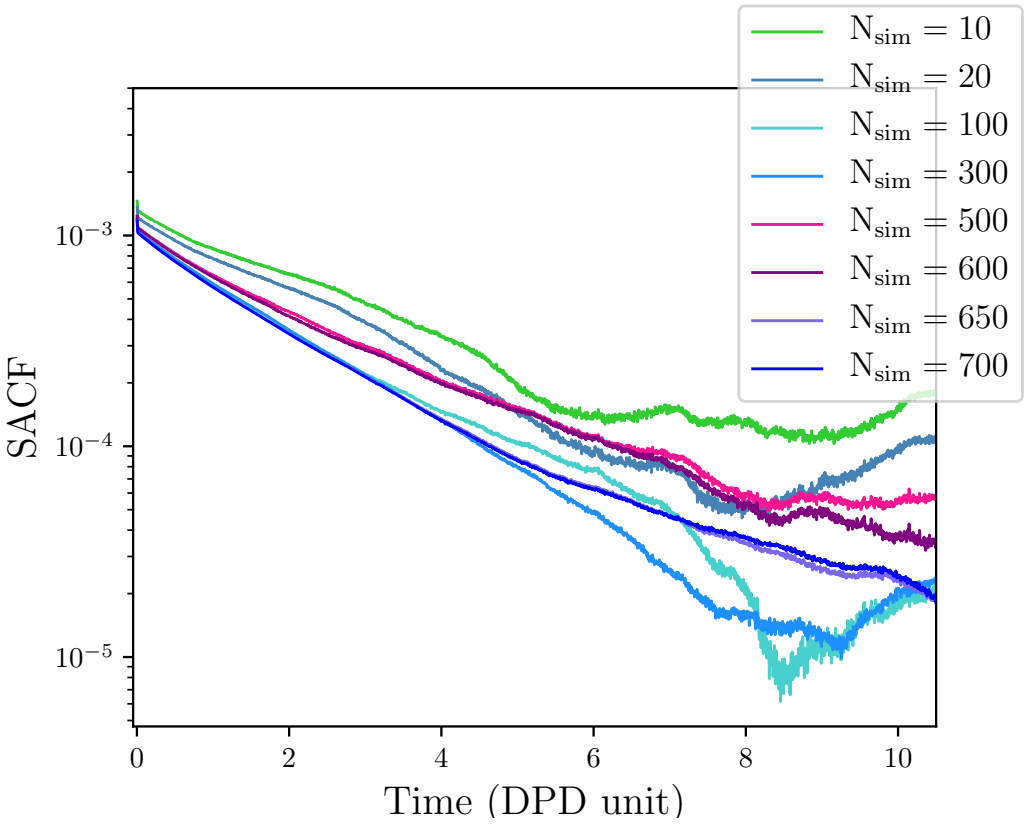

Figure 7: N_{sim}-loop: SACF curves produced for different values of N_{sim} (left) and the correspondent CVI curves (right). The N_{sim}-loop stops at $N_{sim} = 700$.

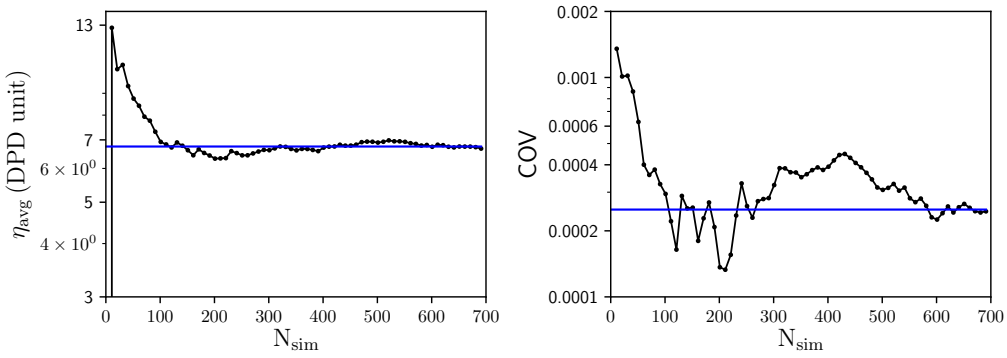

Figure 8: Trend of the mean viscosity value, η_{avg}, and the corresponding coefficient of variation, COV, increasing N_{sim}. Blue lines indicate the values of η_{fin} and COV_{fin} identified by the algorithm.

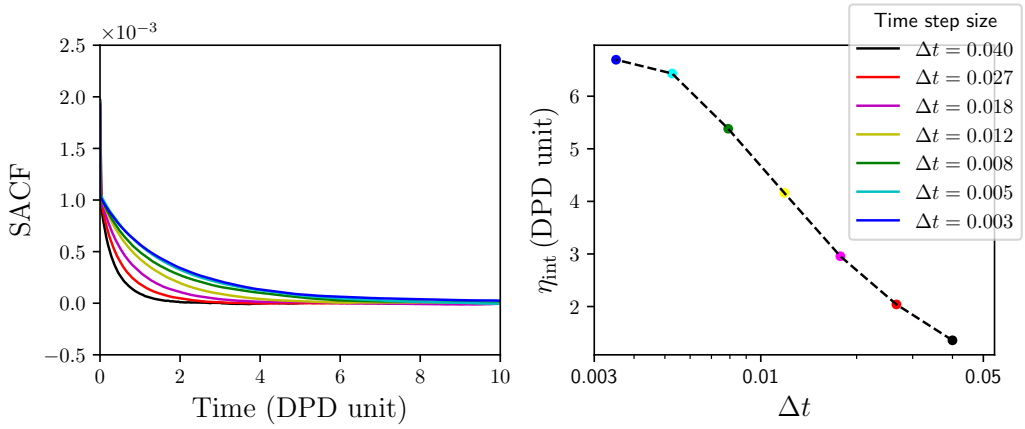

Figure 9: Δt-loop: SACF curves produced for the different time step sizes (left) and the correspondent values of η_{int} computed (right). The Δt-loop stops at $\Delta t = 0.003$ and returns the value of η_{fin} which is the blue one.

since the use of high Δt may lead to large deviation of the system temperature together with large fluctuations that may destroy the structure of the fluid. However, to have a quantitative information about the system we defined an average error as

$$\varepsilon = \frac{\sum_{i=1}^{N_r} |g(r_i) - g^*(r_i)|}{N_r}, \tag{22}$$

where $g^*(r)$ is the RDF computed with the minimum Δt investigated ($\Delta t = 0.003$), N_r is the number of points in which the RDF is evaluated. The ε displayed in Fig. 10 decreases as the time step size decreases. Moreover, the further reduction from $\Delta t = 0.008$ to $\Delta t = 0.005$ produced a marginal variation in the error, showing that the reduction of the time step size plays a greater role in the evaluation of the transport properties rather than the static ones.

4.3. Schmidt number scaling with effective friction coefficient

This paragraph shows the results of the parametric study in which the DPD parameter related to bead friction, s, r_c^D and γ, are varied to reproduce the behavior of different simple fluids with different transport properties. As mentioned in Section 3.3, for each simulation of this parametric study it is possible to calculate the effective friction coefficient γ_{eff} according to Eq. (20), namely a measure of the total friction of the fluid. In Fig. 11 the viscosity, the diffusivity and the Schmidt number are plotted against the effective friction coefficient. The fitting curves

Table 5: Time step size effect on the system temperature.

Δt	$(k_B\mathrm{T})_{\mathrm{avg}}$	$(k_B\mathrm{T})_{\mathrm{std}}$ [%]
0.040	1.0002	0.82
0.027	1.0000	0.84
0.018	0.9998	0.82
0.012	1.0000	0.84
0.008	1.0001	0.80
0.005	1.0001	0.80
0.003	0.9999	0.80

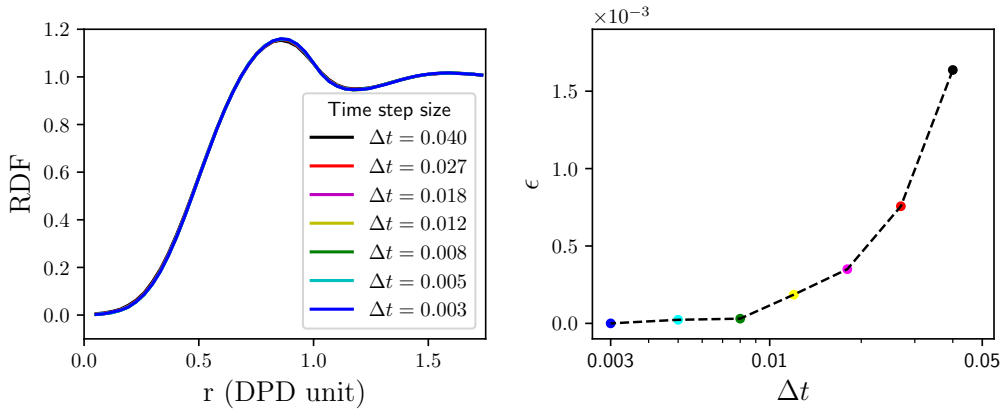

Figure 10: Time step size effect on the RDF (left) and on ε (right). The blue RDF curve (left) corresponds is measured with the Δt value identified by the Δ-loop and the blue point (right) is the ε associated, which is the lowest one.

obtained are reported as red lines in Fig. 11. A simplified relation between the effective friction coefficient and the varied DPD parameters is derived by a multi-parametric fitting leading to the following expression:

$$\gamma_{\text{eff}} \simeq \frac{4\pi}{3}\alpha\frac{\gamma}{s^{0.9}}\rho(r_c^D)^3, \tag{23}$$

where the constant $\alpha = 0.032$ comes from the fitting procedure. The simplified relationship for the γ_{eff}, Eq. (23), is valid for all simple fluids whose potential repulsion parameter a is equal to 25. For different values of a the contribution to γ_{eff} of the variation of the radial distribution function must be taken into account.

While Eq. (23) is the proper way to compute the effective friction coefficient at the end of a simulation, Eq. (23) can be used before running the simulation in order to estimate the effective friction coefficient for a specific set of parameters. In fact, γ_{eff} results proportional to $\frac{4\pi}{3}\rho(r_c^D)^3$ which represents the number of particles interacting via dissipative terms with each particle in the interaction volume. The r_c^D acts on the number of neighbouring beads increasing the number of dissipative interactions. The values of γ and s module the strength of such interactions. The higher γ and the smaller s yields a stronger particle interactions. However, momentum is transported much faster than mass, since momentum can be transferred to other particles by collisions while diffusion is limited to the bead movement. Thus, the viscosity and the diffusivity are expected to, respectively, increase and decrease with increasing γ_{eff}. In this sense, the results reported in Fig. 11 confirm the expectations. In particular, the viscosity η exhibits a linear dependence on the γ_{eff}, i.e.,

$$\eta = 0.0672\gamma_{\text{eff}} + 0.4325. \tag{24}$$

Although for low friction fluids the effect of dissipative contribution to the viscosity is negligible, in the zero-friction limit the viscosity has a no zero value because of the kinetic contribution [50]. While the dissipative contribution to the stress is due to the explicit friction force acting between particles moving on different streamlines, and therefore it is function of γ_{eff}, the kinetic contribution is due to particles diffusing across streamlines and it is not directly related to γ_{eff}. The self-diffusivity was indeed found to be inversely proportional to the γ_{eff} according to the following fitting relationship:

$$\mathscr{D} = 0.2736\gamma_{\text{eff}}^{-1}, \tag{25}$$

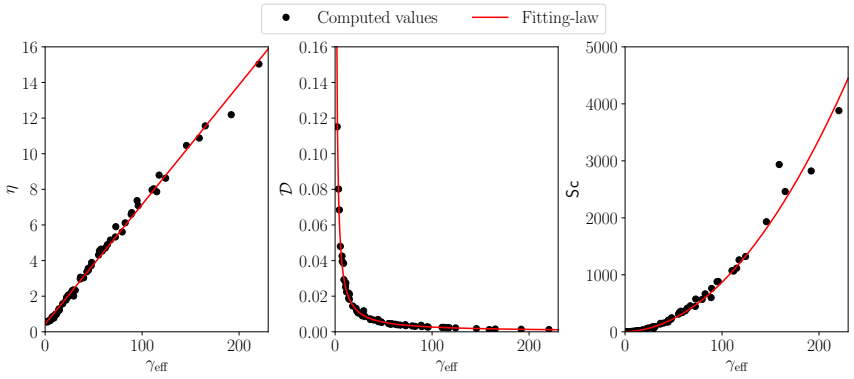

Figure 11: Linear fitting on the viscosity values (left), power-law fitting on the diffusivity values (middle) and the polynomial fitting on the Schmidt number values (right) as a function of the effective friction coefficient parameter. The red lines are to fit the simulation results (black symbols).

meaning that the higher the friction, the lower the diffusivity as expected. The Sc number relationship in function of the γ_{eff} is derived as ratio between Eq. (24) and Eq. (25) resulting in the following:

$$\text{Sc} = 0.0818\gamma_{\text{eff}}^2 + 0.5269\gamma_{\text{eff}}. \tag{26}$$

To asses the model goodness-of-fit, the coefficient of determination (R-squared) and the normalized root mean square error (NRMSE) are reported in Table 6 for all the fitting relations derived in this study. All the fitting relationships recovered present goodness-of-fit measures of the same order of magnitude.

Table 6: Goodness-of-fit measures from the fitting on the effective friction coefficient, the viscosity, the diffusivity the Schmidt number values.

Fitted quantity	R^2	*NRMSE*
γ_{eff}	0.954	0.048
η	0.994	0.017
$\mathscr{D}$	0.876	0.059
Sc	0.978	0.030

The fitting relationships, Eq. (24), Eq. (25) and Eq. (26), were built for values of γ_{eff} ranging from about 1 to 220, covering simple fluids with Sc numbers ranging from approximately 1 to 3880. The Supporting Information reports for each Sc number obtained the corresponding γ_{eff} and the used set of dissipative

parameters s, γ and r_c^D to realize that specific simulation. However, it should be noted that, according to Eq. (23), there are different infinite combinations of these parameters capable of giving the same γ_{eff}. Another aspect to highlight is the fact that the combined use of Eq. (23) and Eq. (26) makes possible to simulate other simple fluids with different transport properties, knowing the Sc number of such fluids. In fact, Eq. (26) can be exploited to obtain the corresponding value of γ_{eff}, given the experimental Sc number. Then Eq. (23) can be used to obtain a set of fixed parameters giving the desired γ_{eff}; fixed two parameters, the third one is found. Fixed the γ_{eff}, η and $\mathscr{D}$ are fixed according to Eq. (24) and Eq. (25). For instance, the parameters needed to properly simulate liquid water at 25°C (Sc $\simeq$ 370 experimentally) are available in the table in correspondence of γ_{eff} equal to 62.137, giving Sc = 369.589. The only limitation is that all the simulations were performed with the same conservative soft-potential, meaning that the isothermal compressibility must be the same of water, which is a reasonable assumption for liquids.

5. Conclusions

In this work, the results obtained by two time integration methods for dissipative particle dynamics (DPD), Shardlow-splitting algorithm (VV-SSA) and modified velocity-Verlet algorithm (VV), were compared for the solution of a DPD model with transverse thermostat with a particular attention to their performance in the evaluation of transport properties. We found that the VV-SSA eliminates the integrator-induced artifacts and performs better in a wider friction region than the VV. Moreover, larger values of the time step can be adopted in transport properties calculation with VV-SSA. The soft interparticle forces neither change abruptly nor diverge at short distances, which permits long integration steps as compared to the atomistic simulations. This fact appreciably accelerate the simulations and, together with the coarse-graining allows for the study of mesoscale phenomena making it appealing for modeling complex systems rheology.

To overcome the numerical issue related to the calculation of the transport quantities from a numerical simulation, we formulated and developed an automated algorithm based on Green-Kubo formula, in which the stress autocorrelation function (SACF) is calculated through an approximated algorithm. Our method automatically finds the parameter of the approximated algorithm for the calculation of the SACF, showing that there is the possibility to get wrong results if these parameters are poorly chosen. Moreover, we found that transport properties can be wrongly evaluated if the time step of the DPD simulation is not

accurately selected. The developed automated method is able to find, for a given DPD system, the viscosity value that it is independent on the choice of time step.

Finally from a parametric study in which we varied all the dissipative parameters of the DPD model, we evaluated the effective friction coefficient, viscosity, self-diffusivity and Sc number. From this dataset made by over 60 simulations, we derived a fitting relation for the effective friction coefficient as a function of different DPD parameters, and fitting relations for viscosity, self-diffusivity and Schmidt (Sc) number as a function of the effective friction coefficient. This analysis shows that infinite sets of DPD parameters may give the same effective friction coefficient, in turn leading to the same Sc number and radial distribution function. In other words, it is possible to simulate the same fluid with the same rheological behavior with different sets of DPD parameters. Moreover, we found that viscosity increases linearly with effective friction coefficient, the self-diffusion is inversely proportional, and the ratio between the viscosity and diffusivity recovered fitting curves results in a very good fit for Sc number values computed from simulations. The findings of this work can be used to simulate simple fluids with different Sc number, since the relation between a specific effective friction coefficient and DPD parameters was found. Moreover, although the method proposed in this work was extensively tested and optimized for simple fluids, it is a tool of general validity with the aim of improving the reliability and computational feasibility of the Green-Kubo approach in DPD simulations. Therefore, it is potentially applicable to both simple and complex fluids. However, the application to more topologically complicated systems is not straightforward, since the intrinsic difficulties of the Green-Kubo approach in DPD are accentuated. Indeed, in the case of polymer systems the computational load required for the viscosity prediction is even higher, since the long-time relaxation requires a larger time interval to trace the whole decorrelation dynamics of the SACF. Although there are some challenges to address, the reliability of the method makes it a promising tool to perform a systematic scaling analysis on the polymer systems dynamical properties, such as viscosity and diffusivity, with polymer concentration and length.

Data availability

More details about the time integration schemes and the simulations performed are provided in the Supporting Information material. The code used for the simulations can be found online at https://github.com/mulmopro/AMTP_DPD.

Author contributions

All authors contributed to the study conception and design. Software programming, data collection and analysis were performed by Nunzia Lauriello. The first draft of the manuscript was written by Nunzia Lauriello and all the authors iteratively corrected and contributed to the final version of the manuscript.

Acknowledgments

The authors gratefully acknowledge Tommaso Maria Ungetti for the contribution to the coding of the algorithm. Computational resources were provided by HPC@POLITO. We acknowledge the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support. The financial support from ICSC (Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing, funded by European Union - NextGenerationEU) is also gratefully acknowledged.

References

[1] F. Schmid, Editorial: Multiscale simulation methods for soft matter systems, Journal of Physics: Condensed Matter 34 (2022) 160401.

[2] M. Karimi, H. Droghetti, D. Marchisio, Multiscale modeling of expanding polyurethane foams via computational fluid dynamics and population balance equation, Macromolecular Symposia 360 (2016) 108–122.

[3] M. Karimi, D. Marchisio, E. Laurini, M. Fermeglia, S. Pricl, Bridging the gap across scales: Coupling CFD and MD/GCMC in polyurethane foam simulation, Chemical Engineering Science 178 (2018) 39–47.

[4] P. Ferkl, M. Karimi, D. Marchisio, J. Kosek, Multi-scale modelling of expanding polyurethane foams: Coupling macro- and bubble-scales, Chemical Engineering Science 148 (2016) 55–64.

[5] A. D. Lavino, N. D. Pasquale, P. Carbone, D. Marchisio, A novel multiscale model for the simulation of polymer flash nano-precipitation, Chemical Engineering Science 171 (2017) 485–494.

[6] P. Español, P. B. Warren, Perspective: Dissipative particle dynamics, The Journal of Chemical Physics 146 (2017) 150901.

[7] R. D. Groot, P. B. Warren, Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation, The Journal of Chemical Physics 107 (1997) 4423–4435.

[8] K. P. Santo, A. V. Neimark, Dissipative particle dynamics simulations in colloid and interface science: a review, Advances in Colloid and Interface Science 298 (2021) 102545.

[9] K. Procházka, Z. Limpouchová, M. Štěpánek, K. Šindelka, M. Lísal, DPD modelling of the self- and co-assembly of polymers and polyelectrolytes in aqueous media: Impact on polymer science, Polymers 14 (2022) 404.

[10] R. Mao, M.-T. Lee, A. Vishnyakov, A. V. Neimark, Modeling aggregation of ionic surfactants using a smeared charge approximation in dissipative particle dynamics simulations, The Journal of Physical Chemistry B 119 (2015) 11673–11683.

[11] H. Droghetti, I. Pagonabarraga, P. Carbone, P. Asinari, D. Marchisio, Dissipative particle dynamics simulations of tri-block co-polymer and water: Phase diagram validation and microstructure identification, The Journal of Chemical Physics 149 (2018) 184903.

[12] R. Pasquino, H. Droghetti, P. Carbone, S. Mirzaagha, N. Grizzuti, D. Marchisio, An experimental rheological phase diagram of a tri-block copolymer in water validated against dissipative particle dynamics simulations, Soft Matter 15 (2019) 1396–1404.

[13] M. Lísal, Z. Limpouchová, K. Procházka, The self-assembly of copolymers with one hydrophobic and one polyelectrolyte block in aqueous media: a dissipative particle dynamics study, Physical Chemistry Chemical Physics 18 (2016) 16127–16136.

[14] X. Ye, B. Khomami, Self-assembly of linear diblock copolymers in selective solvents: from single micelles to particles with tri-continuous inner structures, Soft Matter 16 (2022) 6056–6062.

[15] A. D. Lavino, P. Carbone, D. Marchisio, MARTINI coarse-grained model for poly-ε-caprolactone in acetone-water mixtures, The Canadian Journal of Chemical Engineering 98 (2020) 1868–1879.

[16] G. Boccardo, A. Buffo, D. Marchisio, Simulation of mixing in structured fluids with dissipative particle dynamics and validation with experimental data, Chemical Engineering & Technology 42 (2019) 1654–1662.

[17] A. Boromand, S. Jamali, J. M. Maia, Viscosity measurement techniques in dissipative particle dynamics, Computer Physics Communications.

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

[19] G. Jung, F. Schmid, Computing bulk and shear viscosities from simulations of fluids with dissipative and stochastic interactions, The Journal of Chemical Physics 144 (2016) 204104.

[20] M. Panoukidou, C. R. Wand, P. Carbone, Comparison of equilibrium techniques for the viscosity calculation from DPD simulations, Soft Matter 17 (2021) 8343–8353.

[21] C. Junghans, M. Praprotnik, K. Kremer, Transport properties controlled by a thermostat: An extended dissipative particle dynamics thermostat, Soft Matter 4 (2008) 156–161.

[22] S. Yaghoubi, E. Shirani, A. R. Pishevar, Y. Afshar, New modified weight function for the dissipative force in the DPD method to increase the Schmidt number, Europhysics Letters 110 (2015) 24002.

[23] N. Lauriello, J. Kondracki, A. Buffo, G. Boccardo, M. Bouaifi, M. Lisal, D. Marchisio, Simulation of high schmidt number fluids with dissipative particle dynamics: Parameter identification and robust viscosity evaluation, Physics of Fluids 33 (2021) 073106.

[24] P. J. Hoogerbrugge, J. M. V. A. Koelman, Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics, Europhysics Letters 19 (1992) 155–160.

[25] P. Español, P. Warren, Statistical mechanics of dissipative particle dynamics, Europhysics Letters 30 (1995) 191–196.

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

[27] R. Liuzhen, H. Hu, L. Bao, L. Xie, J. Wen, Transverse effect on liquid viscosity: A many-body dissipative particle dynamics simulation study, Physics of Fluids 34 (2022) 012006–012006.

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

[29] W. Pan, I. V. Pivkin, G. E. Karniadakis, Single-particle hydrodynamics in DPD: A new formulation, Europhysics Letters 84 (2008) 10012.

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

[31] I. Vattulainen, M. Karttunen, G. Besold, J. M. Polson, Integration schemes for dissipative particle dynamics simulations: From softly interacting systems towards hybrid models, The Journal of Chemical Physics 116 (2002) 3967–3979.

[32] B. Leimkuhler, X. Shang, On the numerical treatment of dissipative particle dynamics and related systems, Journal of Computational Physics 280 (2015) 72–95.

[33] P. Nikunen, M. Karttunen, I. Vattulainen, How would you integrate the equations of motion in dissipative particle dynamics simulations?, Computer Physics Communications 153 (2022) 407–423.

[34] T. Shardlow, Splitting for dissipative particle dynamics, Journal on Scientific Computing 24 (2003) 1267–1282.

[35] M. Lísal, J. K. Brennan, J. Bonet Avalos, Dissipative particle dynamics at isothermal, isobaric, isoenergetic, and isoenthalpic conditions using shardlow-like splitting algorithms, The Journal of Chemical Physics 135 (2011) 1–19.

[36] J. P. Larentzos, J. K. Brennan, J. D. Moore, M. Lísal, W. D. Mattson, Parallel implementation of isothermal and isoenergetic dissipative particle dynamics using shardlow-like splitting algorithms, Computer Physics Communications 185 (2014) 1987–1998.

[37] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, S. J. Plimpton, Lammps - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, Computer Physics Communications 271 (2022) 108171.

[38] C. Junghans, M. Praprotnik, K. Kremer, Transport properties controlled by a thermostat: An extended dissipative particle dynamics thermostat, Soft Matter 4 (2008) 156–161.

[39] P. Español, Fluid particle model, Physical Review E 57 (1998) 2930–2948.

[40] L. Verlet, Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules, Physical Review 159 (1967) 98–103.

[41] G. Besold, I. Vattulainen, M. Karttunen, J. M. Polson, Towards better integrators for dissipative particle dynamics simulations, Physical Review E 62 (2000) R7611–R7614.

[42] J.-P. Hansen, I. R. McDonald, Theory of Simple Liquids, Academic Press, Amsterdam, Netherlands, 2013.

[43] M. S. Green, Markoff random processes and the statistical mechanics of time-dependent phenomena. II. irreversible processes in fluids, The Journal of Chemical Physics 22 (1954) 398–413.

[44] R. Kubo, Statistical-mechanical theory of irreversible processes. i. general theory and simple applications to magnetic and conduction problems, Journal of the Physical Society of Japan 12 (1957) 570–586.

[45] M. H. Ernst, R. Brito, New green-kubo formulas for transport coefficients in hard-sphere, langevin fluids and the likes, Europhysics Letters 73 (2006) 183–189.

[46] D. C. Malaspina, M. Lísal, J. P. Larentzos, M. A. D. Brennan, J. K., J. Bonet Avalos, Transport coefficients from Einstein-Helfand relations using standard and energy-conserving dissipative particle dynamics methods, Physical Chemistry Chemical Physics 25 (2023) 12025–12040.

[47] I. Pagonabarraga, M. H. J. Hagen, D. Frenkel, Self-consistent dissipative particle dynamics algorithm, Europhysics Letters 42 (1998) 377.

[48] A. Chaudhri, J. R. Lukes, Velocity and stress autocorrelation decay in isothermal dissipative particle dynamics, Physical Review E 81 (2010) 026707.

[49] E. a. J. F. Peters, Elimination of time step effects in DPD, Europhysics Letters 66 (2004) 311.

[50] C. A. Marsh, G. Backx, M. H. Ernst, Static and dynamic properties of dissipative particle dynamics, Physical Review E 56 (1997) 1676–1691.