



Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

An improved dissipative coupling scheme for a system of Molecular Dynamics particles interacting with a Lattice Boltzmann fluid

Nikita Tretyakov^{a,*}, Burkhard Dünweg^{a,b}^a Max-Planck-Institut für Polymerforschung, Ackermannweg 10, 55128 Mainz, Germany^b Department of Chemical Engineering, Monash University, Clayton, Victoria 3800, Australia

ARTICLE INFO

Article history:

Received 7 October 2016

Received in revised form 18 March 2017

Accepted 27 March 2017

Available online xxx

Keywords:

Lattice Boltzmann

Molecular Dynamics

Fluid-particle coupling

ABSTRACT

We consider the dissipative coupling between a stochastic Lattice Boltzmann (LB) fluid and a particle-based Molecular Dynamics (MD) system, as it was first introduced by Ahlrichs and Dünweg (1999). The fluid velocity at the position of a particle is determined by interpolation, such that a Stokes friction force gives rise to an exchange of momentum between the particle and the surrounding fluid nodes. For efficiency reasons, the LB time step is chosen as a multiple of the MD time step, such that the MD system is updated more frequently than the LB fluid. In this situation, there are different ways to implement the coupling: Either the fluid velocity at the surrounding nodes is only updated every LB time step, or it is updated every MD step. It is demonstrated that the latter choice, which enforces momentum conservation on a significantly shorter time scale, is clearly superior in terms of temperature stability and accuracy, and nevertheless only marginally slower in terms of execution speed. The second variant is therefore the recommended implementation.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

In the last decades the Lattice Boltzmann (LB) technique [1–5] has evolved into a well-founded and efficient numerical tool for the study of fluid mechanics. It has numerous applications, ranging from the studies of turbulence [6] and other macroscopic fluid dynamics problems [7] to soft matter investigations on the meso- or microscale. Hydrodynamics of soft matter is in itself a large field, and LB has been applied to, e. g., liquid crystals [8], two-phase flows [9], binary mixtures [10], and hybrid simulations of particle-based systems, like colloids or polymers, in a solvent. The present paper is a methodological investigation dealing with this last application, which is based upon coupling LB to Molecular Dynamics (MD). This method, which will be referred to by LB/MD, has been described in detail in Ref. [4].

In colloidal dispersions or polymer solutions the molecular structure of the solvent is often irrelevant, while dynamic correlations between the dispersed particles, transmitted via fast momentum transport through the solvent (the so-called “hydrodynamic interactions”) are of paramount importance. There are many ways to take these correlations into account in a simulation, of which LB/MD is only one. Competing approaches are Brownian Dynamics (BD) [11], Dissipative Particle Dynamics (DPD) [12], Multi-Particle Collision Dynamics (MPCD) [13], Smoothed Dissipative Particle

Dynamics (SDPD) [14], and “conventional” Navier–Stokes equation (NSE) solvers [15]. All these methods have advantages and disadvantages, and these have (at least partly) been discussed in Ref. [4]. Important criteria that a simulation method should satisfy are: (i) consistent representation of thermal fluctuations, which are very important on the small length scales of soft matter (satisfied by all); (ii) linear scaling (satisfied by all except BD); and (iii) control over the amplitude of thermal fluctuations which should depend on the degree of coarse-graining (or the length-scale resolution) of the simulation (satisfied only by LB/MD, SDPD, and NSE). LB/MD is particularly attractive for several reasons: (i) due to the lattice, LB is based on a tight data structure, which allows efficient memory management; (ii) due to the streaming-and-collision structure of the algorithm, the method exhibits a high degree of locality, which makes it amenable to parallelization based upon geometric domain decomposition. Indeed, in a comparative study LB/MD was found to be significantly faster than a DPD simulation of the same physical system [16]. A disadvantage of lattice methods is however their inability to deal with difficult boundary conditions, in particular in cases where these involve a deforming simulation cell [17].

It is clear that LB/MD soft-matter simulations have to take care of two aspects that are foreign to the plain LB method, which is essentially not much more than an NSE solver: On the one hand, one has to introduce thermal fluctuations by means of a suitable stochastic collision operator, and on the other hand, one needs a suitable coupling scheme for interaction with the particle-based system. The first aspect has seen significant progress in the

* Corresponding author.

E-mail address: tretyakov@mpip-mainz.mpg.de (N. Tretyakov).

last two decades [18–23], and this topic shall not be our concern here. For the coupling, various schemes have been developed. Among the most prominent methods, one can mention reflecting boundary conditions [18,19,24,4], force coupling [25,4], the immersed boundary method (IBM) [26] and external boundary forces (EBF) [27].

The EBF method is applied to extended objects to satisfy a no-slip boundary condition on their surface. The IBM represents a fluid–particle interface by a set of Lagrangian nodes and interactions are applied as body forces to the fluid. These approaches result in a fairly accurate representation of hydrodynamic boundary conditions, however at the expense of a somewhat complicated algorithm. On the other hand, many soft-matter systems involve objects that are quite large and very “soft” (like polymers of various molecular architectures, tethered membranes, etc.). For these systems the details of the coupling on the local (or monomer) scale do not matter very much – it is only important that the hydrodynamic interactions are correctly represented on larger scales (larger than the monomer size but still significantly smaller than the size of the object as a whole). Therefore, a very simple coupling scheme is desirable, and the force coupling originally put forward by Ahrlich and Dünweg [25] and recently refined by Schiller [28] satisfies this criterion. It is also clear that Ladd’s reflecting boundary method [18,19] is not suitable for polymer systems, since this would require to model each monomer as an extended sphere, which would be computationally much more expensive than the point-particle representation used in force coupling. It is this latter method upon which we will focus in the present paper.

The force coupling algorithm is inherently dissipative, i. e. the velocity of an MD particle is damped with respect to the velocity of the LB fluid interpolated to the particle’s position. Random forces are added to the particles to account for thermal noise. It should be noted that the fluctuation–dissipation theorem stipulates that every dissipation mechanism needs to be compensated by a corresponding noise. This means not only that the viscous damping within the LB fluid must be compensated by a stochastic collision operator, but also that the damping of the particles relative to the surrounding flow needs a compensating noise as well. The counterparts of the coupling forces (damping plus noise) are exerted on the LB fluid to conserve the total momentum. The calculation of the coupling forces takes place every MD step, but the LB update typically needs to be done only after several MD steps. This scheme allows us to capture the dynamics of the fluid and the immersed particles correctly and reproduce hydrodynamic behavior. However, the MD and LB timesteps have to be chosen wisely to find a compromise between the performance and the heat-up of the particle-based system at moderate friction coefficients, which must be viewed as a discretization error. It turns out that the details of this momentum exchange have a significant influence on the size of the discretization error, and the topic of the present investigation is to improve the method with respect to this aspect.

A naive and straightforward approach would involve a recalculation of the streaming velocities at the surrounding LB nodes only every LB step. In the present paper, we investigate both this method as well as a refined one, where the streaming velocities at the surrounding nodes are rather re-calculated every MD time step, in accord with the coupling forces. This latter scheme is obviously more accurate, and gives conservation of total momentum not only on the scale of the LB time step, but rather of the MD time step. We also find that this improves the temperature stability of the simulation substantially, and permits more freedom in the choice of the MD and LB time steps. Furthermore, the computational overhead associated with the improved scheme is insignificant, since it employs already available momentum changes and only adds a few more operations on the surrounding lattice sites.

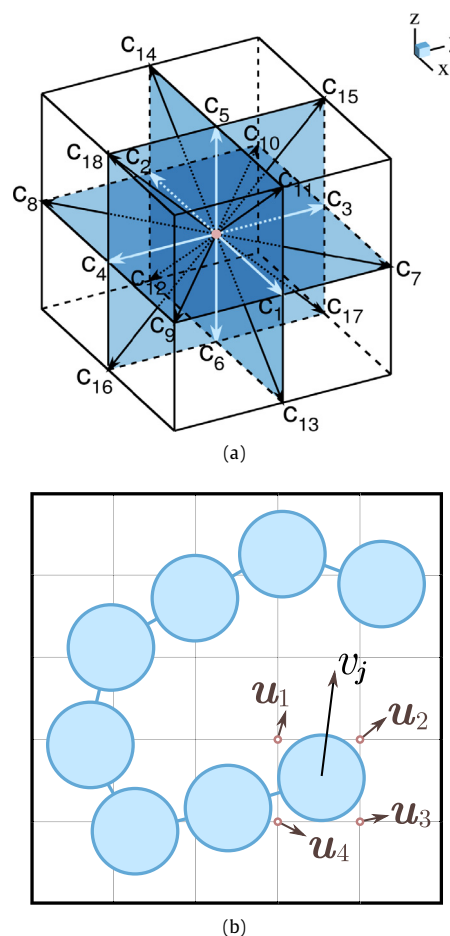


Fig. 1. (a) D3Q19 scheme with 19 velocities connecting a chosen lattice site with its neighbors. (b) Schematic representation of the LB to MD coupling.

The paper is organized as follows: Section 2 presents a short overview of the LB method. The details on the coupling technique and the update scheme are given in Section 3, together with a comparison between the two strategies mentioned above. We conclude in Section 4 with a short summary.

2. Lattice Boltzmann technique

In this section we provide a brief explanation of the LB method. For details and the underlying theory, we refer the reader to the review given in Ref. [4].

The LB scheme can be viewed as a version of coarse-graining of the solvent fluid: Instead of explicit consideration of solvent molecules and their degrees of freedom, the LB method deals with a set of so-called populations $f_i(\vec{r}, t)$ on every lattice site \vec{r} at time t . The population f_i is a quantity proportional to the number of fluid particles flowing with a specific velocity, locally at position \vec{r} at time t . Typically, f_i is interpreted as the local mass density associated with the lattice velocity \vec{c}_i . The finite set of velocities is chosen such that in one time step neighboring lattice sites are connected. The most popular model in three dimensions is called D3Q19. It has 19 velocity vectors \vec{c}_i (including $\vec{c}_0 = 0$), and is schematically shown in Fig. 1a.

The hydrodynamic quantities of the fluid are found by evaluating moments of the populations with respect to the discrete velocity set: The mass density and the mass density flux (or momentum

Download English Version:

<https://daneshyari.com/en/article/4964484>

Download Persian Version:

<https://daneshyari.com/article/4964484>

[Daneshyari.com](https://daneshyari.com)