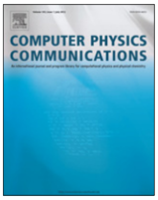
Contents lists available at SciVerse ScienceDirect

Computer Physics Communications

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



Exploiting seeding of random number generators for efficient domain decomposition parallelization of dissipative particle dynamics

Y. Afshar[a,b,*], F. Schmid[b], A. Pishevar[a], S. Worley[c]

[a] Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, 84156-83111, Iran
[b] Institute of Physics, Johannes-Gutenberg Universität Mainz, D55099 Mainz, Germany
[c] Worley Laboratories, United States

ARTICLE INFO

Article history:
Received 23 April 2012
Received in revised form
21 November 2012
Accepted 5 December 2012
Available online 22 December 2012

Keywords:
Random number generator
Saru
Parallel computing
Domain decomposition
Dissipative particle dynamics

ABSTRACT

Dissipative particle dynamics (DPD) is a new promising method commonly used in coarse-grained simulations of soft matter and biomolecular systems at constant temperature. The DPD thermostat involves the evaluation of stochastic or random forces between pairs of neighboring particles in every time step. In a parallel computing environment, the transfer of these forces from node to node can be very time consuming. In this paper we describe the implementation of a seeded random number generator with three input seeds at each step which enables the complete generation of the pairwise stochastic forces in parallel DPD simulations with minimal communication between nodes.

1. Introduction

Dissipative particle dynamics (DPD) is a stochastic coarse-grained simulation model [1] that has been used to study soft matter, complex fluids and biomolecular systems, such as self-assembling amphiphilic micelles and bilayers [2–5], polymers, proteins, nanoparticles, and colloidal systems [6–9]. In the DPD method, a thermostat is provided by dissipative and random forces, which are combined with conservative forces. This thermostat is known to have several favorable properties, *i.e.*, it satisfies Newton's third law by construction and owing to mass, momentum and temperature conservation, hydrodynamics is also correctly reproduced.

In recent years, advances in massively parallel structures have led to dramatic increases in computing performance and many previously untractable scientific computing problems have come within reach. Therefore, there is great interest in developing and accelerating algorithms and methods that improve the computing efficiency on parallel architectures. The transfer of methods that were developed for a serial environment to a parallel environment, where work is distributed over many CPU cores, is not always straightforward. If a method cannot be implemented efficiently in a parallel environment because of communication requirements, speedup can be lost. In DPD simulations, one communication intensive step is the application of the stochastic pairwise forces. Here we propose a strategy which allows one to evaluate these forces with minimal communication between nodes.

In the following section, we briefly introduce the DPD simulation technique, discuss the issue of parallelization, the standard solution, and present our alternative approach. This approach requires the use of a special seeded pseudo random number generator (PRNG). In Section 3, we discuss the issue of PRNGs for parallel simulations, and we introduce very briefly a seeded random number generator with three input seeds which meets our needs. Finally, in Section 4, we provide a performance and scaling analysis of our code by a set of benchmarks and, from this analysis, draw our conclusions in the last Section 5. Details and background on parallelization using domain decomposition, and on the data structures required for implementing the different parallelization strategies are provided in the appendices.

* Corresponding author at: Institute of Physics, Johannes-Gutenberg Universität Mainz, D55099 Mainz, Germany. Tel.: +49 17671254324.
E-mail addresses: ya.afshar@gmail.com, yaser@uni-mainz.de (Y. Afshar), friederike.schmid@uni-mainz.de (F. Schmid), apishe@cc.iut.ac.ir (A. Pishevar), prng@jamasaru.com (S. Worley).

0010-4655/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cpc.2012.12.003

2. Dissipative particle dynamics

Dissipative particle dynamics (DPD) was introduced by Hoogerbrugge and Koelman [10] as a particle based method to simulate complex fluids with the correct hydrodynamics on mesoscopic time and length scales. DPD describes the motion of particles, or beads, which represent the center of mass of a fluid droplet, *i.e.* each DPD particle is a momentum carrying group of molecules. All the degrees of freedom smaller than a bead radius are assumed to have been integrated out, resulting in a coarse-grain representation of fluid elements and in the possibility to adopt soft interactions between the beads.

DPD resembles Molecular Dynamics (MD) in that the particles move in continuous space in discrete time steps, following Newton's laws, but, due to the soft coarse-grained interactions, larger time and length scales are accessible, compared to standard MD. This allows for the study of physical behavior on time scales that can be many orders of magnitude greater than the accessible time scales with MD. In addition to the soft-repulsive (conservative) interaction, DPD models include two other forces: a dissipative force that slows the particles down and removes energy, which can be seen as a viscous resistance, and a stochastic force, which, on average, adds energy to the system and accounts for the degrees of freedom which have been removed by the coarse-graining process. These two forces act together as a thermostat for the system. Each of the three DPD forces is pairwise additive, conserves momentum, and acts along the line joining two particles. The DPD thermostat can also be combined with standard conservative forces, providing a galilean invariant stochastic thermostat for equilibrium or nonequilibrium MD simulations at constant temperature [11].

2.1. Equations of motion

The DPD equations of motion are given by

$$\dot{\mathbf{r}}_i = \frac{\mathbf{P}_i}{m_i} \tag{1}$$

$$\dot{\mathbf{P}}_i = \mathbf{F}_i = \mathbf{F}_i^C + \mathbf{F}_i^D + \mathbf{F}_i^R, \tag{2}$$

where $\mathbf{F}_i^C$, refers to a conservative force, $\mathbf{F}_i^D$ to a dissipative force, and $\mathbf{F}_i^R$ to a stochastic force. The conservative force has two contributions, one describing intermolecular non-bonded interactions and the other describing intramolecular interactions. Most DPD studies express the former contribution by a soft repulsive force in the form

$$\mathbf{F}_{ij}^C = \begin{cases} a_{ij}\left(1 - \dfrac{r_{ij}}{r_c}\right)\mathbf{e}_{ij} & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c, \end{cases} \tag{3}$$

where the coefficient a_{ij} represents the maximum repulsion strength, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, is the distance between particles **i** and **j** ($r_{ij} = |\mathbf{r}_{ij}|$), $\mathbf{e}_{ij} = \frac{\mathbf{r}_{ij}}{r_{ij}}$ and r_c is a cutoff radius which gives the range of the interactions. The dissipative and random forces $\mathbf{F}_i^D$ and $\mathbf{F}_i^R$ have the following form

$$\mathbf{F}_{ij}^D = -\gamma\omega^D(r_{ij})(\mathbf{e}_{ij}\cdot\mathbf{v}_{ij})\mathbf{e}_{ij} \tag{4}$$

$$\mathbf{F}_{ij}^R = \sigma\omega^R(r_{ij})\xi_{ij}\delta t^{-1/2}\mathbf{e}_{ij} \tag{5}$$

where $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ is the velocity difference between particles **i** and **j**. Here γ is a friction coefficient representing viscous damping and σ is the noise amplitude. ξ_{ij} is a random variable, independent for each pair of particles and for each time step, with zero mean and unit variance. $\omega^D(r_{ij})$ and $\omega^R(r_{ij})$ are arbitrary weight functions. The stochastic and dissipative forces are related by a fluctuation–dissipation relation, which implies

$$\begin{aligned} &\omega^D(r) = [\omega^R(r)]^2 \\ &\sigma^2 = 2\gamma K_B T_{\text{ext}}, \end{aligned} \tag{6}$$

where $K_B T_{\text{ext}}$ is the target Boltzmann temperature of the system. The weight function is usually chosen as follows

$$\omega^R(r) = \begin{cases} \left(1 - \dfrac{r}{r_c}\right) & r < r_c \\ 0 & r \geq r_c. \end{cases} \tag{7}$$

The forces acting on a DPD particle **i** are pairwise additive and can be expressed as a sum over all pair forces with neighbor particles within the interaction range. As in MD, the forces on each particle are computed in each time step. The particles are then moved and the force is recomputed. Here we use the velocity-Verlet algorithm of Groot and Warren [12] to integrate the equations of motion.

2.2. Parallelization via domain decomposition

In the present paper we address the issue of parallelizing DPD simulations using domain decomposition (DD) approaches. The basic concept is reviewed in Appendix A. Here we just stress our main point: In DD parallelization schemes, communication arises between nodes if pair forces $\mathbf{F}_{ij} = -\mathbf{F}_{ji}$ act between particles i and j on different nodes. In standard MD simulations with purely conservative interactions, it suffices to transmit the information of the positions of interacting particles. In DPD simulations, more communication is necessary: To evaluate the dissipative interactions, one must also transmit the information on the velocities. To evaluate the stochastic interactions, one must ensure that the same random variables $\xi_{ij} = \xi_{ji}$ are used on both nodes. In the standard implementation, this is done