Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers and Chemical Engineering 29 (2005) 1543–1553
Computers & Chemical Engineering
www.elsevier.com/locate/compchemeng

Parallel implementation of macro-scale pseudo-particle simulation for particle–fluid systems

Xiaowei Wang, Li Guo*, Wei Ge*, Dexiang Tang, Jingsen Ma, Zhangyuan Yang, Jinghai Li

Multi-Phase Reaction Laboratory, Institute of Process Engineering, Chinese Academy of Sciences, P.O. Box 353, Beijing 100080, China

Received 16 September 2003; received in revised form 19 November 2004; accepted 6 December 2004
Available online 17 January 2005

Abstract

As a particle method, macro-scale pseudo-particle modeling (MaPPM) is an effective approach applied to micro-scale simulation of particle–fluid systems. In this paper, a parallel algorithm for macro-scale pseudo-particle modeling based on spatial decomposition (SD) is presented. The parallel implementation utilizes MPI as the programming environment. Due to movement of particles during simulation, the parallelization of MaPPM may suffer from load imbalance and attendant performance degradation. Recursive Coordinate Bisection (RCB) is adopted to partition the whole computational domain in a dynamic fashion to balance the workload in processors. The Shift scheme is modified to meet the communication requirement in the dynamic partition. The parallel approach was applied to simulation of bubble behavior in gas–solid fluidized beds with different system sizes to test its performance. The computations were conducted on cluster of workstations (COW). Experimental results show that the algorithm has a good scalability. With dynamic load balancing (DLB), the parallel efficiency can be improved by up to 8%. To sum up, it was a successful implementation for the parallelization of macro-scale pseudo-particle modeling.

Keywords: Particle–fluid systems; Macro-scale pseudo-particle modeling; Parallel algorithm; Dynamic load balancing

1. Introduction

Particle–fluid system exists widely both in nature and industrial process, such as fluidization systems in chemical industry. However, as a complex system, its mechanism has not been understood sufficiently to meet the requirements of science or technology development. With the dramatic development of computer technology, computer simulation has become an increasingly important tool for the exploration of this complicated mechanism. In the last decades, we have witnessed the rapid growth in new approaches so-called particle method (PM) for computer simulations. They include grid techniques such as LBM-lattice Boltzmann model (Chen & Doolen, 1998; Chopard & Droz, 1998) and LGA-lattice gas automata (Rothman & Zaleski, 1997; Boon, 1991), and meshless particle methods such as MD-molecular dynamics (Rapaport, 1987), DSMC-direct simulation Monte-Carlo (Bird, 1994; Oran, Oh, & Cybyk, 1998), dissipative particle dynamics (DPD; Hoogerbrugge & Koelman, 1992), FPM-fluid particle modeling (Español, 1998) and smoothed particle hydrodynamics (SPH; Gingold & Monaghan, 1977; Libersky, Petschek, Carney, Hipp, & Allahdadi, 1993). Macro-scale pseudo-particle modeling (MaPPM; Ge & Li, 2001), in which fluid is discretized into mesoscale particles, is also a particle method devoted in the simulation of particle–fluid systems.

MaPPM has been applied to the simulation of classical flow field such as plane channel flow, flow around single cylinder, and small-scale fluidization phenomena, for instance, particle clustering and slugging (Ge & Li, 2001, 2002, 2003a). The computational results validated the model when compared with experiments or theoretical analysis, and manifest the unique advantage of MaPPM to incorporate complex boundary conditions. However,

* Corresponding authors. Tel.: +86 10 6262 6705; fax: +86 10 6256 1822.
E-mail addresses: xwwang@home.ipe.ac.cn (X. Wang), lguo@home.ipe.ac.cn (L. Guo), wge@home.ipe.ac.cn (W. Ge).

0098-1354/$ – see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compchemeng.2004.12.006

its wider application is only possible with the advent of high-performance parallel computers taking advantage of its inherent parallelism. Some exploratory work has been done on parallel implementation of the model for single-phase and two-phase simulations (Ge & Li, 2002; Tang et al., 2004), and designs of a general parallel platform for discrete particle simulation (Ge & Li, 2002). The work has indicated that MaPPM can be parallelized on cluster of workstations (COW) with a high efficiency. But at present the computational space is partitioned only with SLICE (Srinivasan, Ashok, Jonsson, Kalonji, & Zahorjan, 1997a, 1997b) scheme that imposes a cap on the number of processors that can be used and affects the parallel granularity and efficiency in most cases except when the flow field is especially long in a certain direction. Here, we present a more flexible parallel algorithm that can partition the computational space with BEAM or BLOCK (Srinivasan et al., 1997a) scheme. The workload imbalance issues in pseudo-particle simulation and the relevant communication scheme are also considered.

The remainder of this paper is organized as follows. In Section 2, we give a brief introduction to MaPPM. In Section 3, we describe the parallel algorithm for MaPPM in detail. In Section 4, experimental results of parallel simulations of bubble behavior in fluidization are presented to demonstrate the performance of our implementation. Discussion of the parallel algorithm is also included in this section. Finally, we conclude with a summary in Section 5.

2. Macro-scale pseudo-particle modeling

Macro-scale pseudo-particle modeling (Ge & Li, 2001, 2003a) is developed from the original pseudo-particle modeling (PPM; Ge & Li, 1999, 2003b) which presents a hybrid of MD and DSMC. The major revision is to upgrade the pseudo-particle interactions to that of mesoscale fluid elements. Similar to smoothed particle hydrodynamics, the basic concept of MaPPM is to express the value of a function (f) on a point (a) as a weighted average of its values on some neighboring points (i), or physically, particles, i.e.

$$f_{\mathrm{a}} = \sum_i f_i \frac{m_i}{\rho_i} W(r_{\mathrm{a}i}) \tag{1}$$

where m and r are particle mass and position, ρ the density there, and $r_{\mathrm{a}i} = |\vec{r}_{\mathrm{a}i}| = |\vec{r}_{\mathrm{a}} - \vec{r}_i|$ (the same for other variables below). Usually the neighborhood is where $r_{ai} = |\vec{r}_{ai}| < R$ and the weight function W is isotropic, we denote $W(r_{\mathrm{a}i})$ as $W_{\mathrm{a}i}$. W should be normalized to satisfy

$$\int_0^R W(r) A(r) \mathrm{d}r = 1 \tag{2}$$

where $A(r) = 2$, $2\pi r$ and $4\pi r^2$ for 1D, 2D and 3D cases, respectively.

But in MaPPM, derivatives of f to different orders are developed in a way just opposite to SPH method, in which the derivatives are directly calculated from Eq. (1). Since f could be any function, it certainly includes its derivatives. So, instead of taking the average and then performing differentiation, we can calculate the directional derivatives by finite difference to neighboring points and take a weighted average of them to express the gradient, and similarly, the Laplacian,

$$\nabla f|_{\mathrm{a}} = D \sum_i \frac{f_{i\mathrm{a}}}{r_{\mathrm{a}i}^2} \vec{r}_{i\mathrm{a}} W_{\mathrm{a}i} \frac{m_i}{\rho_i} \tag{3}$$

$$\Delta \vec{f}|_{\mathrm{a}} = 2D \sum_i \frac{\vec{f}_{i\mathrm{a}}}{r_{\mathrm{a}i}^2} W_{\mathrm{a}i} \frac{m_i}{\rho_i} \tag{4}$$

where D is the number of dimensions of computational space, and $f_{i\mathrm{a}} = f_i - f_{\mathrm{a}}$, $f_{i\mathrm{a}}/r_{i\mathrm{a}}$ is the derivatives expressed in finite differences.

Now we try to express the Navier–Stokes (N–S) equation in MaPPM. For simplicity, we only consider virtually incompressible flows. Moreover, since density variations are smooth, we can assign an identical mass to all particles. We also assume negligible temperature variation (so pressure p is proportional to ρ, i.e. $p = k\rho$) and constant dynamic viscosity μ, which excludes very dense gas whose μ also depends on ρ. Now the N–S equation reduces to

$$\rho \frac{\mathrm{d}\vec{V}}{\mathrm{d}t} = \rho \vec{g} - \nabla p + \mu \Delta \vec{V} \tag{5}$$

where V is particle velocity and g is mass force intensity. For a weakly compressible flow, the overall density difference is usually very small compared with the mean density ρ_m, which simplifies Eq. (5) as

$$\frac{\mathrm{d}\vec{V}}{\mathrm{d}t} = \vec{g} - k' \nabla \rho + v \Delta \vec{V} \tag{6}$$

where $k' = k/\rho_m$ and $v = \mu/\rho_m$.

Meanwhile, from Eqs. (3) and (4) we can rewrite the operators on the right-hand side as:

$$\nabla \rho|_{\mathrm{a}} = D \sum_i \frac{W_{\mathrm{a}i}}{r_{\mathrm{a}i}^2} \vec{r}_{i\mathrm{a}} \tag{7}$$

$$\Delta \vec{V}|_{\mathrm{a}} = \frac{2D}{\rho_m} \sum_i \frac{\vec{V}_{i\mathrm{a}}}{r_{\mathrm{a}i}^2} W_{\mathrm{a}i} \tag{8}$$

where ρ_i is approximated as ρ_m and $m = 1$ is used. Using Eqs. (7) and (8), from Eq. (6) we can get

$$\frac{\mathrm{d}\vec{V}}{\mathrm{d}t}|_{\mathrm{a}} = \vec{g} - k' D \sum_i \frac{W_{\mathrm{a}i}}{r_{\mathrm{a}i}^2} \vec{r}_{i\mathrm{a}} + \frac{2Dv}{\rho_m} \sum_i \frac{\vec{V}_{i\mathrm{a}}}{r_{\mathrm{a}i}^2} W_{\mathrm{a}i} \tag{9}$$

MaPPM has an algorithm similar to MD (Ge & Li, 2000, 2003a). The essence of MaPPM is the numerical integration, over time, of the classical Newtonian equations of position, velocity and force for an ensemble of interacting particles, which are solid particles and pseudo-particles in particle–fluid system. Force computation is the most intensive part in a pseudo-particle simulation. There are three