Journal of Computational Physics 242 (2013) 809–827

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

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

An efficient parallel simulation of interacting inertial particles in homogeneous isotropic turbulence

Ryo Onishi [a,b,*], Keiko Takahashi [a], J.C. Vassilicos [b]

[a] Earth Simulator Center, Japan Agency for Marine-Earth Science and Technology, 3173-25 Showa-machi, Kanazawa-ku, Yokohama, Kanagawa 236-0001, Japan
[b] Department of Aeronautics, Imperial College London, SW7 2AZ, UK

ARTICLE INFO

Article history:
Received 25 April 2012
Received in revised form 10 January 2013
Accepted 10 February 2013
Available online 27 February 2013

Keywords:
Parallel computing
Particle collision
Hydrodynamic interaction
Homogeneous isotropic turbulence

ABSTRACT

This study has conducted parallel simulations of interacting inertial particles in statistically-steady isotropic turbulence using a newly-developed efficient parallel simulation code. Flow is computed with a fourth-order finite-difference method and particles are tracked with the Lagrangian method. A binary-based superposition method has been developed and implemented in the code in order to investigate the hydrodynamic interaction among many particles. The code adopts an MPI library for a distributed-memory parallelization and is designed to minimize the MPI communication, which leads to a high parallel performance. The code has been run to obtain collision statistics of a monodisperse system with $St = 0.4$ particles, where St is the Stokes number representing the particle relaxation time relative to the Kolmogorov time. The attained Taylor-microscale based Reynolds number R_λ ranges from 54.9 to 527. The largest simulation computed the flow on 2000^3 grids and 1000^3 (one billion) particles. Numerical results have shown that the collision kernel increases for R_λ<100 then decreases as R_λ increases. This Reynolds dependency is attributed to that of the radial distribution function at contact, which measures the contribution of particle clustering to the collision kernel. The results have also shown that the hydrodynamic interaction for $St = 0.4$ particles decreases both the radial relative velocity and radial distribution function at contact, leading the collision efficiency less than unity. The collision efficiency increases from 0.65 to 0.75 as R_λ increases for R_λ<200 and then saturates.

1. Introduction

Several mechanisms have been proposed in the literature to explain what causes the fast size-broadening of cloud droplets, which could result in quick rain initiation at the early stage of cloud development. Examples are enhanced collision rate of cloud droplets by turbulence [11,14], turbulence entrainment [6,17], giant cloud condensate nuclei [46,37] and turbulent dispersions of cloud droplets [34]. The most intensely discussed is the first mechanism; enhanced collision rate by turbulence. This has initiated extensive research on particle collisions in turbulence ([36,43,33,25,9, and references therein]).

There are several collision models that predict collision rates of particles in turbulence. Saffman and Turner [32] analytically derived a collision model for particles with zero or very small St (=τ_p/τ_η, where τ_p is the particle relaxation time and τ_η the Kolmogorov time), while Abrahamson [1] derived a model for particles with large St. There is yet no widely accepted model for finite inertial particles, although water droplets in cumulus clouds have finite inertia; cloud droplets have

* Corresponding author at: Earth Simulator Center, Japan Agency for Marine-Earth Science and Technology, 3173-25 Showa-machi, Kanazawa-ku, Yokohama, Kanagawa 236-0001, Japan.
E-mail address: onishi.ryo@jamstec.go.jp (R. Onishi).

0021-9991/$ - see front matter
http://dx.doi.org/10.1016/j.jcp.2013.02.027

$St = O(10^{-2\sim 0})$ and rain drops $St = O(10^{0\sim 2})$. One difficulty arises from the preferential motion of inertial particles. Inertial particles preferentially cluster in regions of low vorticity and high strain if $St \ll 1$ [20], and cluster in a way to mimic the clustering of zero-acceleration points by the sweep-stick mechanism if $1 \lesssim St \lesssim \tau_p/T$, where T is the integral time scale of the turbulence [7]. This matters because clustering increases the mean collision rate [36]. The clustering effect prevents the construction of a fully-analytical model for finite-inertial particles, and requires several empirical parameters in collision models [49,43,48,22,12]. Those parameters are usually determined by direct numerical simulation (DNS) data. Data from laboratory experiments [33,19] would of course help, but available data are very much limited.

One serious problem is that no collision data is available for high Reynolds number flows. The Taylor-microscale based Reynolds number R_λ ($= u'l_\lambda/\nu$, where u' is the rms of velocity fluctuations, l_λ the Taylor microscale and ν the kinematic viscosity) for collision statistics attained by DNS has been at most $R_\lambda \sim 100$. This value is much smaller than those in cloud turbulence, in which R_λ ranges from 10^3 (shallow cumulus clouds) to 10^5 (deep cumulus clouds). Nevertheless, there are several studies where collision models were used in cloud simulations to investigate the impact of enhanced collisions of cloud droplets [24,45,40,26]. They simply extrapolated their collision models to high R_λ, without justification. A simple solution is to obtain collision statistics for high R_λ flows for justifying models, which requires high-performance computing.

Code parallelization is indispensable for high-performance computing. The parallelization is classified into two types. One is the shared-memory parallelization (openMP and auto parallelization libraries are commonly used), and the other the distributed-memory parallelization (message-passing interface, MPI, is commonly used). In the shared-memory parallelization, all processors operate independently but share the same memory resources, i.e., global memory. The global memory concept provides a user-friendly programming perspective to memory. However, shared-memory computers cannot scale very well. Most of them have only ten or fewer processors. In contrast, memory is scalable with number of processors in the distributed-memory parallelization, which therefore is preferable in massively-parallel simulations. Processors have their own local memory and there is no concept of global memory space across all processors. When a processor needs access to data in another processor, data must be communicated through network connecting inter-processor memory, which has much narrower band than that between processor and local memory. Therefore the key to success of massively-parallel simulations with distributed-memory parallelization is in reducing the amount of data communications.

There have been numerous DNS codes for colliding particles in turbulence [43,29,13,5,39,44,25]. One may notice that most of them adopt pseudo-spectral models (PSMs). Unfortunately, few of the PSM codes for particle collisions are designed for the distributed-memory parallel simulations (the only one exception to the authors' knowledge is the very recent work by Rosa et al. [31]), and therefore the attained R_λ has been limited. Furthermore, a parallel PSM code faces major difficulties in massively parallel computing: PSM requires all-to-all data communication for the Fourier transformation. This prevents the PSM from maintaining good parallel efficiency for massively-parallel simulations. Another difficulty is imposed when coupling the PSM with particle calculations. The flow is computed in wavespace in PSM, but Lagrangian particles are in physical space. Code developers therefore need to consider domain decompositions in both wavespace and physical space. These two difficulties could be major reasons why there are few distributed-memory codes for colliding particles employing PSM.

Recently, Onishi et al. [23] developed a finite-difference model (FDM) with an efficient large-scale forcing scheme named reduced-communication forcing (RCF) for statistically-stationary isotropic turbulence. The FDM employs the three-dimensional domain decomposition leading to high parallel efficiency. They also confirmed good reliability of their FDM, which employs a conservative fourth-order finite difference scheme [21]. FDM requires less communications than PSM, and it is, therefore, suitable for massively-parallel computing. Coupling the particle calculation with the FDM can be an alternative to the coupling with the PSM for simulations of inertial particles in high Reynolds number flows.

One important physical process, which has often been neglected due to its high computational cost, is hydrodynamic interactions between particles. These interactions cause particles tend to avoid collisions, thus often leading to a collision efficiency $E_c < 1$. There are several studies which observed that turbulence increases the collision efficiency ([27,39]). However, these studies did not provide data for high Reynolds flows, leaving the Reynolds number dependency of collision efficiency unclear.

Recently, Ayala et al. [3] (hereafter referred to as AGW07) proposed a numerical scheme to consider the hydrodynamic interaction among colliding particles in three-dimensional turbulence. Their scheme is based on the superposition method [28] and adopts the Gauss–Seidel method to iteratively solve a large linear system. It is reportedly feasible to perform a three-dimensional simulation using the scheme, which still requires a huge computational cost for calculating the hydrodynamic interaction. It is preferable to employ a computationally lighter scheme for larger-size computations.

This study aims to develop an MPI parallel code for interacting particles in homogeneous isotropic turbulence (PIPIT, Parallel code for Interacting Particles in homogeneous Isotropic Turbulence) based on FDM coupled with Lagrangian particle calculations, and run the code to obtain data for high-Reynolds flows. The data obtained is used to investigate the Reynolds number dependencies of collision statistics of inertial particles. An efficient scheme, named binary-based superposition method (BiSM), for the hydrodynamic interaction calculation is also proposed and implemented in PIPIT. BiSM is based on the superposition method but is more accurate than the original superposition method [28] and is more computationally efficient than the scheme by Ayala et al. [3]. The cell-index method [2] is adopted in PIPIT for efficient detection of neighboring pairs.

In the following section, we describe numerical procedures for collision detection (subSection 2.1) and hydrodynamic interaction (subSection 2.2). The main frame of PIPIT is then introduced in Section 3, where the algorithm for efficient parallel simulations is described. Numerical results and discussion are mostly presented in Section 4, which consists of