International Journal of Multiphase Flow 35 (2009) 376–388

Contents lists available at ScienceDirect

International Journal of Multiphase Flow

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

A numerically convergent Lagrangian–Eulerian simulation method for dispersed two-phase flows

R. Garg [a], C. Narayanan [b], S. Subramaniam [a,*]

[a] Department of Mechanical Engineering, Center for Computational Thermo-fluids Research, Iowa State University, Ames, IA 50011, USA
[b] ASCOMP GmbH, Technoparkstrasse 1, 8005 Zurich, Switzerland

ARTICLE INFO

Article history:
Received 29 May 2008
Received in revised form 20 November 2008
Accepted 21 December 2008
Available online 14 January 2009

ABSTRACT

In Lagrangian–Eulerian (LE) simulations of two-way coupled particle-laden flows, the dispersed phase is represented either by real particles or by computational particles. In traditional LE (TLE) simulations, each computational particle is assigned a constant statistical weight, which is defined as the expected number of real particles represented by a computational particle. If the spatial distribution of particles becomes highly non-uniform due to particle–fluid or particle–particle interactions, then TLE simulations fail to yield numerically converged solutions due to high statistical error in regions with few particles. In this work, a particle-laden lid-driven cavity flow is solved on progressively refined grids to demonstrate the inability of TLE simulations to yield numerically converged estimates for the mean interphase momentum transfer term. We propose an improved LE simulation (ILE) method that remedies the above limitation of TLE simulations. In the ILE method, the statistical weights are evolved such that the same physical problem is simulated, but the number density of computational particles is maintained near-uniform throughout the simulation, resulting in statistical error that remains nearly constant with grid refinement. The evolution of statistical weights is rigorously justified by deriving the consistency conditions arising from the requirement that the resulting computational ensemble correspond to a statistical description of the same physical problem with real particles. The same particle-laden lid-driven cavity flow is solved on progressively refined grids to demonstrate the ability of ILE simulation to achieve numerically converged estimates for the mean interphase momentum transfer term. The *accuracy* of the ILE method is quantified using a test problem that admits an analytical solution for the mean interphase momentum transfer term. In order to improve the accuracy of numerical estimates of the mean interphase momentum transfer term, an improved estimator is proposed to replace the conventional estimator. The improved estimator results in more accurate estimates that converge faster than those obtained using the conventional estimator. The ILE simulation method along with the improved estimator is recommended for accurate and numerically convergent LE simulations.

1. Introduction

The Lagrangian–Eulerian approach is widely used to simulate dispersed two-phase flows. In this approach the carrier phase is represented by continuous fields in an Eulerian frame of reference, while the dispersed phase is represented by discrete particles[1] in a Lagrangian frame. In two-phase flows with non-negligible mass loading, the mean interphase momentum transfer term cannot be neglected, and two-way coupling effects must be accounted for. The mean interphase momentum transfer term, which is the average force exerted by the particles on the fluid, accounts for the presence of the dispersed phase on the fluid phase. Generally speaking, in two-phase flows there can also be interphase mass and energy transfer, but we consider the simplest case of isothermal particle-laden flow where these are absent. This simple case is used to illustrate the numerical convergence and accuracy of LE simulation methods, but the conclusions are easily generalized to all two-phase flows.

In numerical implementations of the LE method, the numerical estimate[2] of the mean interphase momentum transfer term (or any other mean quantity) at Eulerian grid nodes is obtained using a finite number of particles, leading to *statistical* and *bias* errors (Garg et al., 2007). Statistical error can be reduced either by increasing the number of particles, or by averaging over multiple independent realizations. Bias error is insensitive to the number of independent realizations and becomes zero only in the limit of infinite particles

* Corresponding author.
E-mail addresses: rahul.garg@gmail.com (R. Garg), chidu@ascomp.ch (C. Narayanan), shankar@iastate.edu (S. Subramaniam).

[1] By particle we mean any dispersed-phase element, including solid particles, droplets and bubbles.

[2] We use the term 'estimate' in the statistical sense, just as the sample mean $(1/N)\sum_{i=1}^{N} X^{(i)}$ of a random variable X is an estimate of $\langle X \rangle$. The term estimate is used to only denote the numerical approximation arising from a finite number of samples without implying approximation in any other sense.

0301-9322/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijmultiphaseflow.2008.12.004

(also called the dense data limit). In addition to these errors, a finite number of grid cells and a finite time step leads to the usual *spatial* and *temporal* discretization errors that are encountered in numerical simulations of single-phase flow. The scaling of each of these error contributions—statistical, bias, and discretization error—with variation of numerical parameters determines the numerical convergence characteristics of any LE numerical implementation. Although LE simulations are frequently used to simulate multiphase flows, their numerical convergence and accuracy properties have not been critically examined until recently (Abraham, 1997; Are et al., 2005; Garg et al., 2007).

LE simulation methods such as point–particle DNS—and to a lesser extent, LES—are intended to be used as predictive simulation tools. LE CFD simulations are used to benchmark other simulation approaches, such as Eulerian–Eulerian (EE) two-fluid models (Moreau et al., 2003; Fan et al., 2004; Fan and Fox, 2008). Therefore, establishing numerical convergence of LE simulations is crucial not only for meaningful validation with experimental data, but also for a proper comparison of the modeling error incurred by different choices for sub-models of the interphase mass, momentum or energy transfer terms. A meaningful comparative assessment of sub-models is possible only if the numerical error is negligible compared to the modeling error. It is important to note that numerical convergence is by itself not sufficient to establish the predictive capability of any simulation method. Clearly, establishing *numerical convergence* along with the *accuracy* of LE simulations are necessary before point–particle DNS or LES can be used as predictive tools.

In traditional LE (TLE) simulations (Sundaram and Collins, 1996; Squires and Eaton, 1990; Elghobashi and Truesdell, 1993; Boivin et al., 1998), the dispersed phase is represented either by real particles or by computational particles. If a fixed number of real particles N_p is used to represent the dispersed phase on a grid with total number of grid cells M, then the statistical error in a grid-based estimate of any mean field quantity increases with grid refinement, resulting in a non-convergent LE simulation. This is because as the grid is refined, fewer and fewer particles are available in each grid cell to form the grid-based mean field estimate. Note that for fixed N_p, the nominal number of particles per grid cell $N_{pc} = N_p/M$ decreases as the grid is refined. Therefore the statistical error, which is inversely proportional to the square root of number of particles per cell, increases. This increase in statistical error eventually overwhelms the reduction in spatial discretization error that is achieved by grid refinement. As a result, the total numerical error increases with grid refinement leading to non-converged TLE solutions.

If rather than using real particles, N_c computational particles (with constant statistical weight), such that their total number scales linearly with the number of grid cells ($N_c = N_{pc}M$), are used, then the *nominal* number of particles per grid cell can be maintained constant in a grid-refinement study. For low Stokes number flows where the particle distribution does not develop strong spatial inhomogeneity, it is possible to obtain numerically converged LE solutions using this approach. However, the spatial distribution of particles can be quite non-uniform in particle-laden systems with finite Stokes number. Fig. 1 shows the spatial distribution of particles in lid-driven cavity flow simulation for a Stokes number equal to 0.8. It can be seen that the particles have preferentially concentrated in regions of the flow field with high rate of strain. Therefore, for finite Stokes number, the computational particles also preferentially concentrate just like the real particles. As a result, in regions with few computational particles, the mean field estimates will once again suffer from high statistical error, resulting in non-uniform spatial distribution of statistical error. Based on the above discussion, we identify two major limitations of the TLE simulations: (i) increase in statistical error with grid refinement and (ii) non-uniform spatial distribution of statistical error.

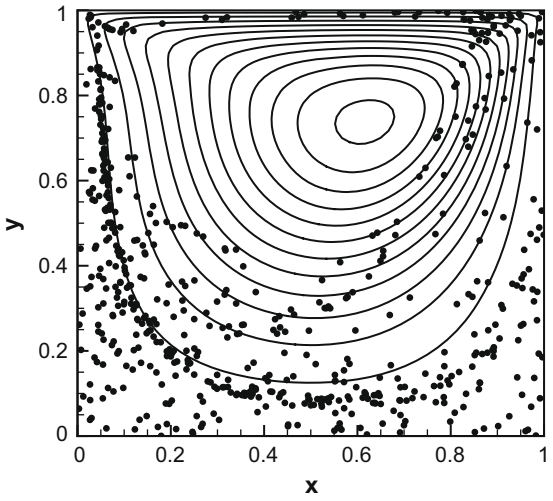

Fig. 1. Snapshot of a one-way coupled lid-driven cavity flow simulation at non-dimensional time tU/L equal to 10. Details are provided in Section 5. The important flow parameters are $Re = UL/\nu = 100, St = \tau_p/\tau_F = 0.8$. The solid lines represent the fluid phase stream function contours and black dots represent the dispersed-phase particles.

The non-convergence of TLE simulations motivates the present work, which aims at developing a numerically convergent and accurate LE simulation method that works robustly for spatially non-uniform particle distributions that arise naturally from the flow physics in time-evolving simulations. In order to overcome the aforementioned limitations of TLE simulations, we propose an improved LE (ILE) simulation technique that uses a computational particle number density control algorithm which is similar to those used in various other particle-based simulations (Pope, 1985; Haworth and Tahry, 1991; Subramaniam and Haworth, 2000; Jaberi et al., 1999; Raman et al., 2005). The computational particle number density control algorithm ensures a near-uniform distribution of computational particles during the entire course of simulation. However, as a result of ensuring near-uniform distribution of computational particles, the statistical weights now need to be evolved in time in order to solve the same physical system. The computational particle number density control procedure relies on the principle of *statistical equivalence* between the TLE (equal and non-evolving statistical weights) and ILE (unequal and time-evolving statistical weights) simulations. It is achieved by annihilating (in case of excess) and cloning (in case of deficient) computational particles in each cell, resulting in nominally equal number of computational particles per cell at all times (Fox, 2003). Thus, the ILE method ensures that the statistical error remains nearly spatially uniform. Therefore, even in the worst case, with increasing grid refinement the statistical error in ILE is guaranteed to remain constant. Note that the total error will decrease with grid refinement at a rate that depends on the order of the spatial discretization scheme, and also on the order of the interpolation scheme used to transfer data between fluid and particle fields. These properties of the ILE method permit a numerically converged LE simulation. It is worthwhile to note that with an efficient parallelization strategy based on domain decomposition, the ILE simulations will be better load-balanced than the TLE simulations.

In earlier work (Garg et al., 2007), we characterized the numerical convergence properties of four interpolation schemes for mean interphase momentum transfer term used in LE simulations in a series of static test problems. The estimation of the mean interphase momentum transfer involves the use of an interpolation scheme in conjunction with an estimator formula. Since those static test problems were designed such that the mean interphase

momentum transfer term could be solved analytically, we were also able to quantify the accuracy of the interpolation schemes. We observe that accurate estimation of the mean interphase momentum transfer term using certain interpolation schemes requires very high numerical resolution: on the order of 100 particles per Eulerian grid cell and 100 independent realizations. This observation motivates the development of more accurate estimators. In this article we evaluate the numerical accuracy of two types of estimators: the conventional estimator (Sundaram and Collins, 1996; Boivin et al., 1998; Narayanan et al., 2002; Patankar and Joseph, 2001; Snider et al., 1998), and an improved estimator. We show that the improved estimator results in more accurate estimates of the mean interphase momentum transfer term than the conventional estimator, and these estimates also converge at a faster rate.

In order to test the numerical convergence and accuracy of the ILE method we consider two problems. The first problem is the particle-laden lid-driven cavity flow shown in Fig. 1, which is representative of a practical two-phase flow system. We solve the particle-laden lid-driven cavity flow on progressively refined grids using the ILE method. We demonstrate that ILE, unlike TLE, yields numerically converged estimates for the mean interphase momentum transfer term.

While numerical convergence of an LE simulation can be characterized for the particle-laden lid-driven cavity problem, we cannot quantify the accuracy of the LE solution because we do not know the exact solution. Therefore, for testing accuracy we extend the static particle test of Garg et al. (2007) to a time-evolving test problem where the particles naturally assume a non-uniform spatial distribution due to the flow physics. Our time-evolving test problem mimics the conditions of real particle-laden flows, and yet is simple enough to permit analytical solution for mean fields like the number density and the interphase momentum transfer term. In order to quantify the accuracy of both TLE and ILE simulation methods, we then solve the test problem on progressively refined grids. Since the mean interphase momentum transfer term is analytically known for the test problem, the accuracy of both the simulations can be easily quantified. We show that by using the ILE simulation with the improved estimator it is possible to obtain *numerically converged* LE simulations with demonstrable *accuracy*.

2. Governing equations

In the LE approach, the dispersed phase consisting of N_p real particles is represented in a Lagrangian frame at time t by $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t), i = 1, \ldots, N_p(t)\}$, where $\mathbf{X}^{(i)}(t)$ denotes the ith particle's position, and $\mathbf{V}^{(i)}(t)$ represents its velocity. For simplicity we consider monodisperse particles here, although the conclusions of this work hold for polydisperse cases as well. The position and velocity of the physical particles evolve by

$$\frac{d\mathbf{X}^{(i)}}{dt} = \mathbf{V}^{(i)}, \tag{1}$$

$$\frac{d\mathbf{V}^{(i)}}{dt} = \frac{\mathbf{f}_p^{(i)}}{m_p^{(i)}} = \mathbf{A}^{(i)}, \quad i = 1, \ldots, N_p(t), \tag{2}$$

where $\mathbf{f}_p^{(i)}$ and $\mathbf{A}^{(i)}$ are, respectively, the instantaneous force and acceleration experienced by the ith physical particle having mass $m_p^{(i)}$.

In order to compute the mean momentum transferred from the particles to the fluid, a statistical description of the particle ensemble is needed to average over all possible particle configurations and velocities. At the single particle level, this statistical description is given by the one-particle distribution function $f(\mathbf{x}, \mathbf{v}, t)$ of kinetic theory, also referred to as the droplet distribution function (ddf) in the context of sprays (Williams, 1958). The density $f(\mathbf{x}, \mathbf{v}, t)$ is related to the position and velocity of the physical particles by

$$f(\mathbf{x}, \mathbf{v}, t) \equiv \langle f'(\mathbf{x}, \mathbf{v}, t) \rangle = \left\langle \sum_{i=1}^{N_p(t)} f_i'(\mathbf{x}, \mathbf{v}, t) \right\rangle = \left\langle \sum_{i=1}^{N_p(t)} \delta_{\mathbf{X}^{(i)}} \delta_{\mathbf{V}^{(i)}} \right\rangle, \tag{3}$$

where f' is the fine-grained density function, f_i' is the fine-grained density function for the ith particle, $\delta_{\mathbf{X}^{(i)}} = \delta(\mathbf{x} - \mathbf{X}^{(i)}(t))$, $\delta_{\mathbf{V}^{(i)}} = \delta(\mathbf{v} - \mathbf{V}^{(i)}(t))$, and the expectation is over all possible particle configurations and velocities of the multiparticle system. The unnormalized density $f(\mathbf{x}, \mathbf{v}, t)$ is not a probability density function (Subramaniam, 2000) because it integrates to the expected total number of particles $\langle N_p \rangle$.

The evolution of the particle system by Eqs. (1) and (2) implies an evolution equation for $f(\mathbf{x}, \mathbf{v}, t)$ (Subramaniam, 2001), which is

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x_k}[v_k f] + \frac{\partial}{\partial v_k}[\langle A_k | \mathbf{x}, \mathbf{v}; t \rangle f] = 0. \tag{4}$$

In the above equation $\langle A_k | \mathbf{x}, \mathbf{v}; t \rangle$ is the expected acceleration conditional on the location $[\mathbf{x}, \mathbf{v}]$ in the position-velocity space, which is defined as

$$\langle A_k | \mathbf{x}, \mathbf{v}; t \rangle = \frac{1}{f(\mathbf{x}, \mathbf{v}, t)} \left\{ \left\langle \sum_{i=1}^{N_p(t)} A_k^{(i)} f_i'(\mathbf{x}, \mathbf{v}, t) \right\rangle \right\}, \quad \text{if} \quad f > 0, \tag{5}$$

and zero otherwise. The closure for this conditional acceleration term is obtained by assuming an acceleration model that includes all the relevant forces arising from particle–particle interactions (e.g., collisional, electrostatic), and particle–fluid interactions (e.g., drag, Saffman lift, added mass, Basset history term). In this work we choose a physical setup where only the drag force is needed to model this term, and a general form that subsumes different drag force correlations is

$$\mathbf{A}^{*(i)}(t) = \mathbf{A}\left(\mathbf{U}^f\left(\mathbf{X}^{(i)}(t), t\right), \mathbf{V}^{(i)}, \rho_f, \nu_f, \rho_p, D_p\right), \tag{6}$$

where $\mathbf{U}^f(\mathbf{X}^{(i)}(t), t)$ is the fluid velocity at the particle location, ρ_f and ν_f are the fluid thermodynamic density and kinematic viscosity, respectively, and ρ_p and D_p are the particle density and diameter, respectively. Even though only drag force is considered in this study, the conclusions will hold equally well for all particle–fluid interactions, with minor modifications to the above functional form of $\mathbf{A}^*$ in order to account for the changes necessitated by the additional physics.

The mean momentum conservation equation in the fluid phase obtained by ensemble-averaging (Drew and Passman, 1998) is

$$\rho_f \alpha_f \left(\frac{\partial \langle \mathbf{U}^f \rangle}{\partial t} + \langle \mathbf{U}^f \rangle \cdot \nabla \langle \mathbf{U}^f \rangle \right) = \nabla \cdot \langle \boldsymbol{\tau} \rangle - \langle \mathbf{F}^{\mathrm{fp}} \rangle + \nabla \cdot \boldsymbol{\tau}^{RS}, \tag{7}$$

where α_f is the average fluid volume fraction, $\boldsymbol{\tau}^{RS}$ is the residual stress resulting from ensemble averaging, and the angle brackets represent phasic averages of the terms. The mean interphase momentum transfer term, $\langle \mathbf{F}^{\mathrm{fp}} \rangle$, that appears in the fluid-phase mean momentum conservation equation is obtained from $\langle \mathbf{f} | \mathbf{x}, \mathbf{v}; t \rangle$, the conditional expectation of the force acting on the physical particles, as follows:

$$\langle \mathbf{F}^{\mathrm{fp}} \rangle (\mathbf{x}, t) = \int_{[\mathbf{v}]} \langle \mathbf{f} | \mathbf{x}, \mathbf{v}; t \rangle f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \tag{8}$$

where the integration is performed over $\mathbf{v}$, the sample space variable corresponding to the particle velocity $\mathbf{V}$.

The dependence of the mean interphase momentum transfer on configuration of the particles, and on the particle velocity distribution, is revealed by decomposing the density $f(\mathbf{x}, \mathbf{v}, t)$ as a product of the particle number density, $n_p(\mathbf{x}, t)$, and the particle conditional velocity pdf, $f_{\mathbf{V}}^c(\mathbf{v} | \mathbf{x}; t)$ (Subramaniam, 2001):

$$f(\mathbf{x},\mathbf{v},t) = n_p(\mathbf{x},t) f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t). \tag{9}$$

Spatial non-uniformity in the particle position distribution manifests itself in the particle number density $n_p(\mathbf{x},t)$, which for non-aggregating particles evolves by (Subramaniam, 2001)

$$\frac{\partial n_p(\mathbf{x},t)}{\partial t} + \nabla\cdot\left\{\langle \mathbf{V}(\mathbf{x},t)\rangle n_p(\mathbf{x},t)\right\} = 0, \tag{10}$$

where $\langle \mathbf{V}(\mathbf{x},t)\rangle$ is the mean particle velocity field. If there is no inflow and outflow, such as in the particle-laden lid-driven cavity flow, then the evolution equation for expected total number of particles ($\langle N_p(t)\rangle = \int_{\mathbf{x}} n_p(\mathbf{x},t)d\mathbf{x}$) becomes

$$\frac{\partial\langle N_p(t)\rangle}{\partial t} = 0. \tag{11}$$

In fact, in this special case the total number of particles N_p (not just the mean $\langle N_p\rangle$) is always constant. From Eqs. (4) and (10), the evolution equation for $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)$ (Subramaniam, 2001) is

$$\begin{aligned}\frac{\partial f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)}{\partial t} &+ \frac{\partial}{\partial x_k}\left[v_k f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)\right] + \frac{\partial}{\partial v_k}\left\{\langle A_k|\mathbf{x},\mathbf{v};t\rangle f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)\right\}\\ &= f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)\frac{\partial \ln n_p(\mathbf{x},t)}{\partial x_k}\{\langle \mathbf{V}(\mathbf{x},t)\rangle - v_k\} + f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)\\ &\quad\times\frac{\partial\langle V_k(\mathbf{x},t)\rangle}{\partial x_k}.\end{aligned} \tag{12}$$

As noted earlier, in LE simulations the physical system described by Eqs. (1) and (2) can be simulated with N_p real particles, or with N_c computational particles. Both simulations constitute an indirect solution of Eq. (4), or equivalently, of Eqs. (10) and (12). In the latter case, the computational ensemble is *statistically equivalent* to the physical system. However, even simulations with real particles can be conveniently interpreted as a special case of statistical equivalence between the computational ensemble and the physical system. Statistical equivalence is ensured by enforcing consistency at all times between

(i) the number density implied by the computational ensemble and the number density corresponding to the physical system, which evolves by Eq. (10), and
(ii) the particle velocity distribution implied by the computational ensemble and the particle velocity distribution corresponding to the physical system, which evolves by Eq. (12).

Any changes to the computational ensemble, such as allowing the statistical weights to evolve in time, must preserve this statistical equivalence with the physical system. Therefore, we describe the statistical equivalence between the computational ensemble and the physical system in some detail in the following section.

3. Statistical description of dispersed phase: computational particles

In this section, the number density and particle velocity distribution implied by the ensemble of statistically weighted computational particles are established. The section is sub-divided based on the type of statistical weights used: constant (TLE simulations) or time-evolving (ILE simulation). The consistency requirements for statistical equivalence are derived by equating the number density and particle velocity distribution implied by the computational ensemble to their counterparts in the physical system.

3.1. Traditional LE simulation (TLE): equally weighted particles

In LE simulations, the dispersed-phase consisting of N_p real particles is indirectly represented by N_c computational particles. These N_c computational particles are represented in a Lagrangian frame at time t by $\{\mathbf{X}_c^{(i)}(t), \mathbf{V}_c^{(i)}(t), W^{(i)},\quad i = 1,\ldots,N_c(t)\}$, here $\mathbf{X}_c^{(i)}(t)$ denotes the ith computational particle's position, $\mathbf{V}_c^{(i)}(t)$ its velocity, and $W^{(i)}$ its statistical weight. The statistical weight is defined as the average number of real particles represented by a computational particle. The summation of statistical weights, over all computational particles, equals the expected total number of real particles

$$\sum_{i=1}^{N_c(t)} W^{(i)} = \langle N_p(t)\rangle. \tag{13}$$

The position and velocity of the computational particles evolve by

$$\frac{d\mathbf{X}_c^{(i)}}{dt} = \mathbf{V}_c^{(i)} \tag{14}$$

$$\frac{d\mathbf{V}_c^{(i)}}{dt} = \mathbf{A}_c^{(i)},\quad i = 1,\ldots,N_c \tag{15}$$

where $\mathbf{A}_c^{(i)}$ is the instantaneous acceleration experienced by the ith computational particle. Using a condensed notation $\delta_{\mathbf{X}_c^{(i)}} = \delta(\mathbf{x}-\mathbf{X}_c^{(i)}(t))$ and $\delta_{\mathbf{V}_c^{(i)}} = \delta(\mathbf{v}-\mathbf{V}_c^{(i)}(t))$, it is convenient to define the fine-grained density for the ith computational particle $h_i'(\mathbf{x},\mathbf{v},t) = W^{(i)}\delta_{\mathbf{X}_c^{(i)}}\delta_{\mathbf{V}_c^{(i)}}$. The fine-grained density for the ensemble of N_c computational particles is then written as $h'(\mathbf{x},\mathbf{v},t) = \sum_{i=1}^{N_c(t)} h_i'(\mathbf{x},\mathbf{v},t)$. Analogous to the density function $f(\mathbf{x},\mathbf{v},t)$, which was defined earlier for the real particles, a weighted density function $h(\mathbf{x},\mathbf{v},t)$ for the computational particles is defined in terms of h' as

$$\begin{aligned}h(\mathbf{x},\mathbf{v},t) &\equiv \langle h'(\mathbf{x},\mathbf{v},t)\rangle = \left\langle \sum_{i=1}^{N_c(t)} h_i'(\mathbf{x},\mathbf{v},t)\right\rangle\\ &= \left\langle \sum_{i=1}^{N_c(t)} W^{(i)}\delta_{\mathbf{X}_c^{(i)}}\delta_{\mathbf{V}_c^{(i)}}\right\rangle.\end{aligned} \tag{16}$$

The validity of using computational particles in place of real particles rests on the equivalence between h and f at all time. For the present case, the statistical weight $W^{(i)} = \langle N_p\rangle/N_c$ is equal for each computational particle, and does not evolve. As a result $h = f$ at initial time, and if the acceleration models for real and computational particles are identical (i.e., $\mathbf{A}_c^{(i)} \equiv \mathbf{A}^{*(i)}$), then this equivalence holds between the two statistical descriptions (i.e., $h = f$) for all time.

3.2. Improved LE simulation (ILE): unequal and evolving weights

The improved LE simulation employs N_c computational particles that are also represented in a Lagrangian frame at time t by $\{\mathbf{X}_c^{(i)}(t), \mathbf{V}_c^{(i)}(t), W^{(i)}(t),\quad i = 1,\ldots,N_c(t)\}$. The principal difference between ILE and TLE is that the statistical weight $W^{(i)}(t)$ is now a function of time in ILE. The position and velocity of the computational particles evolve by Eqs. (14) and (15), respectively. The statistical weights evolve by

$$\frac{dW^{(i)}(t)}{dt} = -\Omega^{(i)}(t)W^{(i)}(t),\quad i = 1,\ldots,N_c(t), \tag{17}$$

where $\Omega^{(i)}$ represents the fractional rate of change of statistical weight.

The weighted density function $h(\mathbf{x},\mathbf{v},t) = \left\langle \sum_{i=1}^{N_c(t)} W^{(i)}(t)\delta_{\mathbf{X}_c^{(i)}}\delta_{\mathbf{V}_c^{(i)}}\right\rangle$, which is similar to the definition in Eq. (16), except that here the statistical weights $W^{(i)}(t)$ are not constant but evolve in time. Similar to the decomposition of f in Eq. (9), h is decomposed as

$$h(\mathbf{x},\mathbf{v},t) = \widetilde{n}_p(\mathbf{x},t)\widetilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t), \tag{18}$$

where $\widetilde{h}_{\mathbf{V}_c}^c$ is the conditional velocity pdf of computational particles (the counterpart of $f_{\mathbf{V}}^c$) and $\widetilde{n}_p(\mathbf{x},t)$ is the physical number density

implied by the present statistical description. The implied physical number density $\widetilde{n}_p(\mathbf{x},t)$, which is obtained by integrating the density function h over velocity space, can be expressed as the product of the computational particle number density $n_c(\mathbf{x},t)$ and the conditional expectation of statistical weights $\langle W|\mathbf{x};t\rangle$:

$$\widetilde{n}_p(\mathbf{x},t) = \int_{[\mathbf{v}]} h d\mathbf{v} = \left\langle \sum_{i=1}^{N_c(t)} W^{(i)}(t)\delta\left(\mathbf{X}_c^{(i)}(t)-\mathbf{x}\right)\right\rangle = n_c(\mathbf{x},t)\langle W|\mathbf{x};t\rangle, \tag{19}$$

where the conditional expectation of the statistical weights is defined as

$$\langle W|\mathbf{x};t\rangle = \frac{\left\langle W^{(i)}(t)\delta\left(\mathbf{X}_c^{(i)}(t)-\mathbf{x}\right)\right\rangle}{n_c(\mathbf{x},t)}, \quad \text{if} \quad n_c>0, \tag{20}$$

and is equal to zero otherwise, and $n_c(\mathbf{x},t) = \left\langle \sum_{i=1}^{N_c(t)}\delta\left(\mathbf{X}_c^{(i)}(t)-\mathbf{x}\right)\right\rangle$.

We seek to guarantee the equivalence between h and f at all time by comparing evolution equations for h, and also the fundamental quantities $(\widetilde{n}_p, \langle N_p(t)\rangle, h^c_{\mathbf{V}_c})$ with their counterparts in the physical system. The evolution equation for h is

$$\frac{\partial h}{\partial t} + \underbrace{\langle \Omega|\mathbf{x},\mathbf{v};t\rangle h} + \frac{\partial}{\partial x_k}[v_k h] + \frac{\partial}{\partial v_k}\left\{\langle A_{c,k}|\mathbf{x},\mathbf{v};t\rangle h\right\} = 0, \tag{21}$$

where $\langle \Omega|\mathbf{x},\mathbf{v};t\rangle$ is the conditional expectation of fractional rate of change of statistical weight, which is given by

$$\langle \Omega|\mathbf{x},\mathbf{v};t\rangle = \frac{1}{h(\mathbf{x},\mathbf{v},t)}\left\langle \sum_{i=1}^{N_c}\left\{\Omega^{(i)}h_i'(\mathbf{x},\mathbf{v},t)\right\}\right\rangle, \quad \text{if} \quad h>0 \tag{22}$$

and equal to zero otherwise. The conditional expectation of the acceleration term $\langle A_{c,k}|\mathbf{x},\mathbf{v};t\rangle$ is similarly defined.

The evolution equation for the number density $\widetilde{n}_p$, obtained by integrating Eq. (21) over $\mathbf{v}$ space is

$$\frac{\partial \widetilde{n}_p(\mathbf{x},t)}{\partial t} + \nabla\cdot\left\{\langle \mathbf{V}_c(\mathbf{x},t)\rangle \widetilde{n}_p(\mathbf{x},t)\right\} = \underbrace{\langle \Omega|\mathbf{x};t\rangle \widetilde{n}_p(\mathbf{x},t)}, \tag{23}$$

where the conditional expectation $\langle \Omega|\mathbf{x};t\rangle$ is defined as

$$\langle \Omega|\mathbf{x};t\rangle = \frac{\left\langle \sum_{i=1}^{N_c}\Omega^{(i)}W^{(i)}(t)\delta_{\mathbf{x}_c^{(i)}}\right\rangle}{\widetilde{n}_p(\mathbf{x},t)}, \quad \text{if} \quad \widetilde{n}_p>0 \tag{24}$$

and equal to zero otherwise. The evolution equation for the total number of particles $\langle N_p(t)\rangle$, obtained by integrating Eq. (23) over $\mathbf{x}$ space is

$$\frac{\partial\langle N_p(t)\rangle}{\partial t} = -\underbrace{\langle N_p(t)\rangle\langle\Omega(t)\rangle}, \tag{25}$$

where $\langle\Omega(t)\rangle$ is the unconditional expectation of Ω, which is given as

$$\langle\Omega(t)\rangle = \frac{\left\langle\sum_{i=1}^{N_c}\Omega^{(i)}W^{(i)}(t)\right\rangle}{\langle N_p(t)\rangle}. \tag{26}$$

From the evolution equations for h (Eq. (21)) and number density $\widetilde{n}_p$ (Eq. (23)), the evolution equation for $\widetilde{h}^c_{\mathbf{V}_c}$ can be obtained as

$$\begin{aligned}&\frac{\partial \widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)}{\partial t} + \frac{\partial}{\partial x_k}\left[v_k\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)\right]\\ &+\frac{\partial}{\partial v_k}\left\{\langle A_{c,k}|\mathbf{x},\mathbf{v};t\rangle\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)\right\}\\ &=\underbrace{-\langle\Omega|\mathbf{x},\mathbf{v};t\rangle\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)+\langle\Omega|\mathbf{x};t\rangle\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)}+\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)\\ &\times\frac{\partial\langle V_{c,k}(\mathbf{x},t)\rangle}{\partial x_k}+\widetilde{h}^c_{\mathbf{V}_c}(\mathbf{v}|\mathbf{x};t)\frac{\partial\ln\widetilde{n}_p(\mathbf{x},t)}{\partial x_k}\{\langle\mathbf{V}_c(\mathbf{x},t)\rangle - v_k\}.\end{aligned} \tag{27}$$

3.2.1. Consistency requirements

In the above evolution equations (Eqs. (21), (23), (25), and (27)) for $h, \widetilde{n}_p, \langle N_p(t)\rangle$, and $h^c_{\mathbf{V}_c}$, the underbraced quantities are the extra terms that appear when compared with the corresponding evolution equations (Eqs. (4), (10), (11), and (12)) for the real particles. Comparing the evolution equations for h in TLE, the computational particles have constant statistical weights, and the equivalence of the computational ensemble with the statistical description based on the real particles is trivially verified. For ILE with time-evolving weights, the same equivalence is guaranteed only if the extra term $\langle\Omega|\mathbf{x},\mathbf{v};t\rangle$ that appears in the evolution equation for h (Eq. (21) as compared to Eq. (4) for f) is zero. This automatically then guarantees equivalence of the corresponding number density ($\widetilde{n}_p \equiv n_p$) and velocity PDF's ($\widetilde{h}^c_{\mathbf{V}_c} \equiv f^c_{\mathbf{V}}$). In summary, the computational particles and the real particles are statistically equivalent if the conditional (Eqs. (22) and (24)), and unconditional (Eq. (26)) expectations of fractional rate of change of statistical weight Ω are all zero. These conditions on Ω are consistency requirements, i.e., a prescription of $\Omega^{(i)}(t)$ in Eq. (17) that satisfies these conditions guarantees that the evolution of computational particles by Eqs. (14), (15) and (17) corresponds to the evolution of the physical system as given by Eq. (4). A particle number density control algorithm that ensures a near-uniform spatial distribution of computational particles, and also satisfies all the consistency requirements, is described in the next section.

3.2.2. Computational particle number density control algorithm

To maintain nearly uniform computational particle number density, we use a variant of a commonly used approach in other particle-based numerical methods, such as, PDF methods for turbulent flows (Pope, 1985; Haworth and Tahry, 1991; Subramaniam and Haworth, 2000), direct simulation Monte Carlo methods (Kannenberg and Boyd, 2000), large eddy simulations of turbulent flows using filtered density function approach (Jaberi et al., 1999; Raman et al., 2005). The numerical simulation begins with some initial computational particles that are uniformly distributed in the flow domain. The same statistical weight is assigned to all particles in a cell, with the spatial distribution of statistical weights obeying Eq. (19) with $\tilde{n}_p = n_p$, the specified physical number density. In the ideal case, one would want to maintain a constant number of particles (denoted by $N^{\mathrm{T}}_{\mathrm{pc}}$) in each cell throughout the course of simulation. We find that requiring constant number of particles in each cell is a very stringent requirement, but allowing the number of particles in each cell to lie within some range centered around the ideal value of $N^{\mathrm{T}}_{\mathrm{pc}}$ is a better alternative. In our simulations the minimum number of particles in each cell is specified to be $0.5N^{\mathrm{T}}_{\mathrm{pc}}$, while the maximum number of particles allowed in each cell is $2.0N^{\mathrm{T}}_{\mathrm{pc}}$. After evolving the position and the velocity of all particles by a time step, the number of computational particles in each cell is computed. If this number lies outside the interval $[0.5N^{\mathrm{T}}_{\mathrm{pc}}, 2.0N^{\mathrm{T}}_{\mathrm{pc}}]$, the following actions are taken:

1. $N_{\mathrm{pc}} > 2.0N^{\mathrm{T}}_{\mathrm{pc}}$: In this case, the particle with the lowest statistical weight is annihilated or deactivated, and its weight is equally re-distributed among the remaining particles in the same cell. This annihilation procedure continues until the number of particles in that cell reduces to the desired value of $2N^{\mathrm{T}}_{\mathrm{pc}}$.
2. $N_{\mathrm{pc}} < 0.5N^{\mathrm{T}}_{\mathrm{pc}}$: In this case, the particle with the highest statistical weight is cloned or split into two equally weighted new particles that are randomly placed in the same cell. The new particles retain the properties of the cloned particle[3] such as,

[3] There is nothing unique about this prescription, but it is the simplest approach to preserve a minimum statistical equivalence, at the level of first moments of the density functions f and h, following the splitting procedure. More sophisticated and complex algorithms would be needed to ensure consistency at the second (or higher) moments.

velocity, temperature, etc. This cloning procedure continues until the number of particles in that cell exceeds the minimum desired value of $0.5N_{pc}^{T}$.

In the following we show that this number density control algorithm satisfies the consistency requirements as described in Section 3.2.1. Since the algorithm ensures that the sum of statistical weights of all the computational particles is unchanged, it satisfies the first consistency requirement, $\langle \Omega(t) \rangle = 0$. Since both the annihilation and cloning procedures are performed at the cell level, the second consistency requirement, $\langle \Omega | \mathbf{x}; t \rangle = 0$, is also satisfied. Finally, since the number density control algorithm does not depend on the velocities of the computational particles, the third consistency requirement, $\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle = 0$, that expectation of the fractional rate of change of the statistical weights, conditional on the physical and the velocity spaces, should be zero, is also satisfied. It is important to note that identical evolution equations, given by Eqs. (14) and (15), for particle position and velocity are solved in TLE and ILE. However, in ILE, the particle weights evolve in time as described above. Essentially this corresponds to a specification of $W^{(i)}(t)$ that evolves according to Eq. (17), but we omit the formal mathematical definition of $\Omega^{(i)}(t)$ in favor of the easily understood algorithm.

4. Numerical estimation of mean interphase momentum transfer term

The numerical estimate for the mean interphase momentum transfer, $\{\mathbf{F}^{\mathrm{fp}}(\mathbf{x})\}$, at the mth grid node is obtained as

$$\{\mathbf{F}_m^{\mathrm{fp}}\} = \frac{1}{\mathscr{V}_m} \sum_{i=1}^{N_c(t)} \mathbf{f}_c^{(i)} W^{(i)} K(\mathbf{X}_c^{(i)}, \mathbf{x}^m), \tag{28}$$

where $\mathbf{f}_c^{(i)} = m_p \mathbf{A}_c^{(i)}$ is the force acting on the ith computational particle, $K(\mathbf{X}_c^{(i)}, \mathbf{x}^m)$ is a generic kernel with compact support that determines the influence of the particle force at $\mathbf{X}_c^{(i)}$ on a grid node located at $\mathbf{x}^m$, and $\mathscr{V}_m$ is the geometric volume of the mth grid cell. In the convention followed, $\{\cdot\}$ represents the numerically estimated mean field, while $\langle \cdot \rangle$ represents the analytical mean field. Four interpolation schemes for calculation of the mean interphase momentum transfer term are considered in this work: fourth-order Lagrange polynomial interpolation (LPI-4), second-order Lagrange polynomial interpolation (LPI-2), piecewise cubic approximation (PCA), and a two-stage estimation algorithm (TSE). The details of the interpolation schemes are provided in Garg et al. (2007).

For a 1-D grid, shown in Fig. 2, the numerical estimate for $\left\{F_{x,m}^{\mathrm{fp}}\right\}$, from Eq. (28), at the mth grid node for an $\mathscr{O}$th-order interpolation scheme is

$$\left\{F_{x,m}^{\mathrm{fp}}\right\}^{\mathrm{CE}} = \frac{1}{\mathscr{V}_m} \sum_{v=m-\mathscr{O}/2}^{m+\mathscr{O}/2-1} \sum_{k=1}^{N_c^v} f_x^k W^k b_l^x(\xi_l^k), \tag{29}$$

where $l = m - v + \mathscr{O}/2, b_l^x$ is the basis function at the elemental coordinate ξ_l^k, N_c^v is the number of computational particles in the vth cell, f_x^k is the x component of the force $\mathbf{f}_c^{(k)}$ acting on the kth particle, and the superscript 'CE' stands for conventional estimator. In the above equation, the basis functions are numbered from left to right. For example, if a particle is located in 5th cell (i.e. $v = 5$), then the fourth-order LPI-4 interpolation scheme will yield four non-zero basis functions, b_1 through b_4, which correspond to grid nodes 4 through 7, respectively. The conventional estimator has been extensively used in past LE simulations (Sundaram and Collins, 1996; Boivin et al., 1998; Narayanan et al., 2002; Patankar and Joseph, 2001; Snider et al., 1998).

Here we propose an improved estimator to compute the mean interphase momentum transfer term as

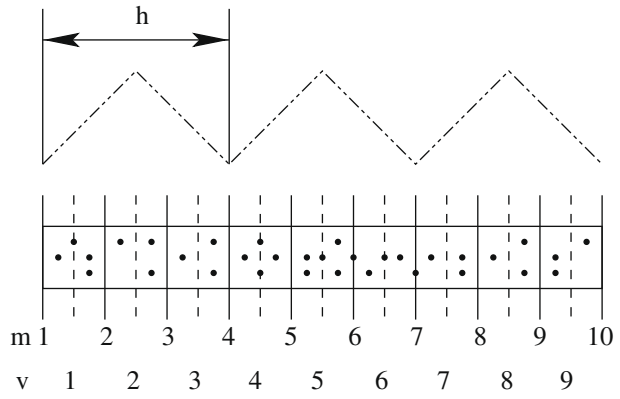

Fig. 2. Schematic of a 1-D grid with dispersed phase particles shown by black dots. Solid and dashed vertical lines, indexed by m, show coarse and fine grids, respectively. The angled intersecting lines on the top represent a typical top hat kernel having bandwidth equal to h.

$$\left\{F_{x,m}^{\mathrm{fp}}\right\}^{\mathrm{IE}} = \frac{1}{\mathscr{V}_m} \frac{\sum_{v=m-\mathscr{O}/2}^{m+\mathscr{O}/2-1} \sum_{k=1}^{N_c^v} \phi_x^k b_l^x(\xi_l^k)}{\sum_{v=m-\mathscr{O}/2}^{m+\mathscr{O}/2-1} \sum_{k=1}^{N_c^v} b_l^x(\xi_l^k)}, \tag{30}$$

where ϕ_x^k is a scaled force acting on the kth particle in cell v, and superscript 'IE' stands for improved estimator. For the kth particle belonging to the vth cell, ϕ_x^k is

$$\phi_x^k = f_x^k \sum_{j=1}^{N_c^v} W^j = f_x^k N_p^v, \tag{31}$$

where N_p^v is the number of physical particles in the vth cell. On substituting the above expression for ϕ_x^k into the expression for improved estimator (Eq. (30)), we get

$$\left\{F_{x,m}^{\mathrm{fp}}\right\}^{\mathrm{IE}} = \frac{1}{\mathscr{V}_m} \sum_{v=m-\mathscr{O}/2}^{m+\mathscr{O}/2-1} \sum_{k=1}^{N_c^v} f_x^k W^{\star k} b_l^x(\xi_l^k), \tag{32}$$

where $W^{\star k}$, the effective statistical weight associated with the kth particle is

$$W^{\star k} = \frac{\sum_{j=1}^{N_c^v} W^j}{\sum_{v=m-\mathscr{O}/2}^{m+\mathscr{O}/2-1} \sum_{k=1}^{N_c^v} b_l^x(\xi_l^k)} = \frac{N_p^v}{\{N_{c,m}\}}. \tag{33}$$

In the above expression, $\{N_{c,m}\}$ is the effective number of computational particles at the mth grid node. Therefore, $W^{\star k}$ can be interpreted as the locally averaged statistical weight. The expressions for the conventional (Eq. (29)) and the improved estimators (32) are very similar except for the difference in the weighting factor. Whereas in the conventional estimator, the weighting factor is simply the statistical weight of the particle, in the improved estimator, the weight factor is a locally averaged value given by Eq. (33).

This improved estimator is similar to the first stage approximation in the TSE interpolation scheme used in earlier studies (Dreeben and Pope, 1992; Subramaniam and Haworth, 2000) and extensively tested for accuracy in Garg et al. (2007). Therefore, the TSE interpolation scheme is always implemented with the improved estimator, while the other three interpolation schemes (LPI-2, LPI-4, and PCA) can be implemented with either the conventional or the improved estimator. Unless otherwise noted, the improved estimator is used to obtain all the results that follow. It will be shown later in a test problem that the improved estimator yields more accurate and faster converging estimates than the conventional estimator for the mean interphase momentum transfer term.

5. Lid-driven cavity flow problem

We first solve the one-way coupled, lid-driven cavity flow problem using both the traditional and the improved LE simulation methods. The carrier fluid momentum conservation equation (Eq. (7)) is solved for primitive variables using the fractional time– stepping procedure of Kim and Moin (1985). Fourth-order accurate Runge–Kutta scheme is used to advance the particle's position and velocity. Second-order accurate central-differencing scheme is used for both the convection and the diffusion terms. The LPI-4 interpolation scheme is used to interpolate the fluid velocity field to the particle location, and the LPI-2 interpolation scheme is used to form the estimates for the mean interphase momentum transfer term.

The carrier flow Reynolds number $Re = L_{\text{ref}}U_{\text{ref}}/\nu_f$, based on the cavity length L_{ref} and lid velocity U_{ref}, is equal to 100. The physical system is a volumetrically dilute particle-laden flow with large particle to fluid density ratio ($\rho_p \gg \rho_f$). The solid particles are monodisperse and small compared to the smallest flow length scale, but large enough so Brownian motion of the particles can be neglected. The Reynolds number for relative motion between the particle and the fluid is $\mathcal{O}(1)$. Under these conditions, the interphase momentum transfer is due to the drag and buoyancy forces. If we neglect buoyancy and assume a linear drag model (which is valid for Reynolds number $\mathcal{O}(1)$), the modeled particle acceleration $\mathbf{A}^{*(i)}$ is given by

$$\mathbf{A}^{*(i)} = \frac{\mathbf{U}^f\left(\mathbf{X}^{(i)}, t\right) - \mathbf{V}^{(i)}}{\tau_p}, \tag{34}$$

where $\tau_p = \rho_p D_p^2/(18\mu_f)$ is the particle momentum response time. The particle Stokes number $St = \tau_p/\tau_f$, which is based on a flow time scale $\tau_f = L_{\text{ref}}/U_{\text{ref}}$, is equal to 0.8. The volume fraction of the dispersed phase α_p is equal to 0.005, resulting in approximately 5300 real particles. The physical problem is solved on progressively refined grids, ranging from the coarsest resolution of 50×50 grid cells to the finest resolution of 100×100 grid cells.

For the TLE simulation, two different approaches are used. In the first approach, referred to as TLE1, real particles (≈ 5300) are used. In the second approach, referred to as TLE2, computational particles with equal and non-evolving statistical weights are used. In TLE2, for all the grid sizes a fixed number of computational particles per cell, $N_{\text{pc}}(t=0) = 20$, are uniformly seeded at the beginning of the simulation.

For the ILE simulation, the computational particles are initially seeded as in TLE2, i.e., $N_{\text{pc}}(t=0) = 20$. The target number of computational particles in each cell N_{pc}^{T} is set equal to 20. Therefore, according to the particle number density control algorithm outlined earlier, the minimum and the maximum number of computational particles per cell are 10 and 40, respectively.

The global error in estimating the mean interphase momentum transfer term is defined as root mean square of the relative error, or

$$\epsilon_F = \frac{1}{M}\sqrt{\sum_{m=1}^{M}\left(\frac{\{\mathbf{F}_m^{\text{fp}}\} - \left\langle \mathbf{F}_m^{\text{fp}}\right\rangle}{\left\langle \mathbf{F}_m^{\text{fp}}\right\rangle}\right)^2}, \tag{35}$$

where $M = M_x M_y M_z$ is the total number of grid cells. In the absence of an analytical solution for the mean interphase momentum transfer term in the current problem, the ILE solution on a highly resolved 150×150 grid is taken to be the reference solution for the purpose of error calculation. The relative root mean square error ϵ_F for each grid is calculated by substituting the interpolated value of reference solution for $\left\langle \mathbf{F}_m^{\text{fp}}\right\rangle$ in the above equation.

The particle-laden lid-driven cavity problem is simulated for 10 non-dimensional time units ($t^* = t/\tau_f$). Fig. 1 shows a snapshot of the fluid stream function field (represented by contour lines) and the dispersed-phase particles (represented by black dots) obtained from the TLE1 simulation. Fig. 3 compares the convergence characteristics of the root mean square relative error ϵ_F, with grid spacing $h = \sqrt{\Delta x \Delta y}$, for different simulations—TLE1, TLE2, ILE. Lines are a simple fit to the data. It is observed that both the TLE simulations, TLE1 and TLE2, fail to yield numerically converged estimates for the mean interphase momentum transfer term. On the other hand, the root mean square relative error for the ILE simulation shows a monotonic decrease, indicating numerical convergence. Although the lid-driven cavity flow results demonstrate the inability of the TLE simulations to yield numerically converged solutions, it is not possible to quantify the accuracy of different simulations in the absence of an analytical solution for the mean interphase momentum transfer term. To address the issue of accuracy, a simple test problem that admits an analytical solution for the mean interphase momentum transfer term is proposed in the next section.

6. Test problem

Here we propose a novel test problem that mimics the conditions of real particle laden flows, and yet is simple enough to permit analytical solution for mean fields like the number density and the mean interphase momentum transfer term. The physical system implied by the test problem admits the same assumptions made for the lid-driven cavity flow, i.e., volumetrically dilute, $\rho_p \gg \rho_f$, monodisperse particles, and Reynolds number for relative motion between the particle and the fluid is $\mathcal{O}(1)$. Therefore, the linear drag model given by Eq. (34) is valid here also.

As represented by the schematic in Fig. 4, a frozen two-dimensional fluid velocity field

$$U_1^f(x,y) = U_0, \tag{36}$$

$$U_2^f(x,y) = U_0\left(1 - \frac{y}{\mathscr{L}_y}\right), \tag{37}$$

is chosen in a domain $\mathscr{D} = [0, \mathscr{L}_x] \times [0, \mathscr{L}_y]$. For this fluid velocity field, the flow time scale τ_f is defined to be equal to $\mathscr{L}_y/U_0$, and the particle Stokes number is $St = \tau_p/\tau_f$.

Particles are injected at $x = 0$, with velocity $\mathbf{V} = (V_1, V_2) = (U_0, 0)$. The particle position and velocity equations (Eqs. (14) and (15)) can be reduced to two second-order ordinary differential

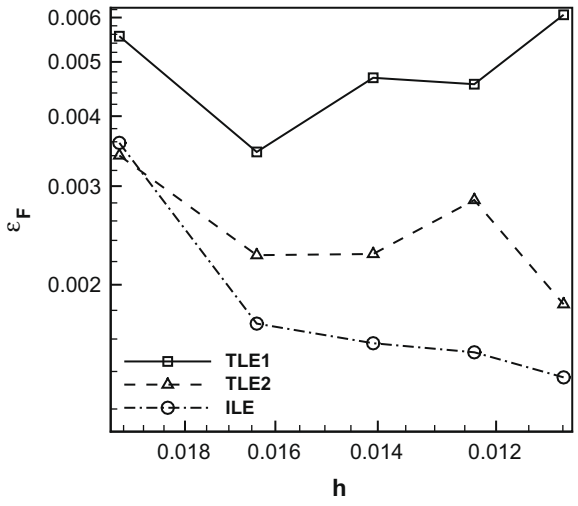

Fig. 3. Comparison of the convergence characteristics of the root mean square relative error ϵ_F with grid spacing h for TLE1, TLE2, and ILE simulations of the lid-driven cavity flow. Lines are a simple fit to the data points.

equations by substituting the frozen fluid velocity field (Eqs. (36) and (37)) into the particle acceleration model (Eq. (34)). These ODE's for the particle trajectory can be solved for any Stokes number. However, depending on the nature of the roots (real distinct or complex conjugate) of the characteristic equation corresponding to these ODEs, two different solutions are possible. Real and distinct roots arise when $St < 0.25$ and complex conjugate roots arise when Stokes number is greater than 0.25. Since preferential concentration is observed at $St \sim 1$ and the resulting spatial distribution of particles will be most demanding of LE simulation methods (cf. Fig. 1 for lid-driven cavity simulation at $St = 0.8$), we choose to solve for the $St > 0.25$ case. For complex conjugate roots, the analytical expressions for the particle trajectory and velocity in y-direction are

$$X_2(t,X_{2,0}) = \mathrm{e}^{-t/2\tau_p}\left[X_{2,0} - \mathscr{L}_y\right]\left\{\cos\left(\frac{\zeta t}{2\tau_p}\right) + \frac{1}{\zeta}\sin\left(\frac{\zeta t}{2\tau_p}\right)\right\} + \mathscr{L}_y, \tag{38}$$

and

$$V_2(t,X_{2,0}) = \frac{2U_0\left[1 - \frac{X_2}{L_y}\right]\sin\left(\frac{\zeta t}{2\tau_p}\right)}{\zeta\left\{\cos\left(\frac{\zeta t}{2\tau_p}\right) + \frac{1}{\zeta}\sin\left(\frac{\zeta t}{2\tau_p}\right)\right\}}, \tag{39}$$

where $X_2(t,X_{2,0})$ and $V_2(t,X_{2,0})$ denote the position and velocity at time t, respectively, of the dispersed phase particle that is located at $X_{2,0}$ at time $t = 0$. The parameter $\zeta = \sqrt{4St - 1}$. Since the particle moves with a constant velocity (i.e., $V_1(t,X_{1,0}) = U_0$) in the x direction, its x coordinate at any time is given by $X_1(t,X_{1,0}) = X_{1,0} + U_0 t$.

The Eulerian mean velocity field for particle phase is denoted $\mathbf{U}^p(\mathbf{x},t)$, and it can be deduced from the Lagrangian solution $\mathbf{V}(t,\mathbf{X}_0)$ (Eq. (39)) by the transformation

$$\mathbf{U}^p(\mathbf{X}(t,\mathbf{X}_0),t) = \mathbf{V}(t,\mathbf{X}_0). \tag{40}$$

The particles are injected based on a specified inlet particle volume fraction field. Since we are interested in a non-uniform number density distribution, we choose a simple transcendental inflow volume fraction of the form

$$\alpha_p(x=0,y) = \frac{\alpha_{p,\max} + \alpha_{p,min}}{2} + \frac{\alpha_{p,\max} - \alpha_{p,\min}}{2}\sin\left(\frac{2\pi y}{\mathscr{L}_y}\right), \tag{41}$$

where $\alpha_{p,\min}$ and $\alpha_{p,\max}$ ensure bounded volume fraction ($0 < \alpha_p < 1$) for all values of y. Given the analytical expressions for the particle trajectory (Eq. (38)), it is straightforward to write down the volume fraction field at any later time. For our test problem, the steady dispersed-phase volume fraction field is

$$\alpha_p(x,y) = \begin{cases} \alpha_p(x=0,Y^{-1}(y))e^{\hat{x}}/g(\hat{x}) & : \ 0 < x \leqslant \mathscr{L}_x, \\ & \quad \mathscr{L}_y\{1 - e^{\hat{x}}g(\hat{x})\} < y \leqslant \mathscr{L}_y; \\ 0 & : \ \text{otherwise}, \end{cases} \tag{42}$$

where $Y^{-1}(y)$ is an inverse function obtained by re-expressing $X_{2,0}$ in Eq. (38) in terms of X_2, such that

$$Y^{-1}(y) \equiv X_{2,0} = \frac{y - \mathscr{L}_y}{g(\hat{x})}\mathrm{e}^{\hat{x}} + \mathscr{L}_y, \tag{43}$$

where $\hat{x} = x/2\tau_p U_0$, and $g(\hat{x}) = \cos(\zeta\hat{x}) + \zeta^{-1}\sin(\zeta\hat{x})$.

For monodisperse particles, the number density field corresponding to this particle volume fraction field is $n_p(x,y) = \alpha_p(x,y)/\mathscr{V}_p$, where $\mathscr{V}_p$ is the particle volume. Fig. 5 shows the contour plot of the normalized analytical mean number density field $n_p(x,y)/n_{p\max}$. From the contour plot, and from the above expression for volume fraction field (Eq. (42)), it is noted that in the region bounded by $0 < x \leqslant \mathscr{L}_x$ and $0 \leqslant y \leqslant \mathscr{L}_y\{1 - e^{\hat{x}}g(\hat{x})\}$, the number density is zero, i.e. no physical particles could be present in this region. From hereon, this curve will be referred to as the bounding streamline.

Similarly, the analytical expression for the mean interphase momentum transfer term obtained from Eq. (8) is

$$\langle \mathbf{F}^{\mathrm{fp}} \rangle(\mathbf{x}) = \frac{m_p n_p(\mathbf{x})}{\tau_p}\left[\mathbf{U}^f(\mathbf{x}) - \mathbf{U}^p(\mathbf{x})\right]. \tag{44}$$

Fig. 5 shows the contour plot of the scaled analytical mean interphase momentum transfer term in y-direction obtained after substituting the fluid velocity field (Eqs. (36) and (37)), number density field, and mean particle velocity field (Eq. (40)) into Eq. (44). Since the particles are injected with $V_1 = U_0$, they experience zero drag in the x direction.

The objective of the test problem is to quantify the accuracy of ILE, TLE1, and TLE2 simulations. Errors in LE simulations arise from: (1) using a finite grid to represent and evolve the fluid velocity field, (2) forward-interpolating the fluid velocity field represented at grid nodes to particle location for calculating particle forces (cf. Eq. (34)), (3) evolution of particle position and velocity using a finite time step, and (4) estimation of mean fields, like the number density or the mean interphase momentum transfer term, from a finite number of particles. The first three sources of error are common to ILE, TLE1, and TLE2. Since the principal difference between the simulation methods is in step (4), the goal is to minimize or eliminate all sources of error, except the backward estimation error (4). Since the fluid velocity field is analytically specified, error (1) due to finite grid size is zero. Specified fluid velocity field also eliminates error (2) due to forward interpolation. A highly accurate, fourth-order Runge–Kutta scheme is used to evolve the position and velocity of the particles in all the tests. Thus, the first two sources of error are totally eliminated, and the third one is minimized.

6.1. Computation setup

The physical domain $\mathscr{D}$ is discretized using a structured grid into $M_x \times M_y \times M_z$ cells. In all our tests, the domain is a unit cube

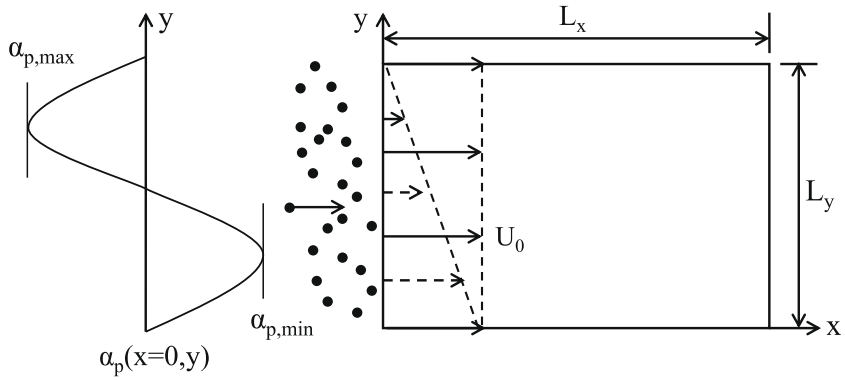

Fig. 4. Schematic of the test problem. Solid line vectors represent U_1^f, dashed line vectors represent U_2^f. and particles, injected at $x = 0$, are shown as black dots.

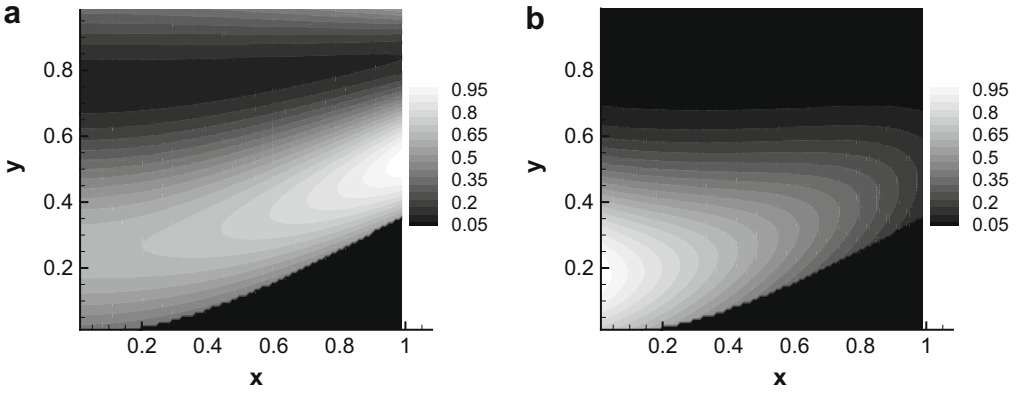

Fig. 5. Contour plots of scaled analytical mean (a) number density $\langle n_p \rangle / \langle n_p \rangle_{\max}$ and (b) interphase momentum transfer term $\langle F_y^{\mathrm{fp}} \rangle / \langle F_y^{\mathrm{fp}} \rangle_{\max}$ for the test problem.

with $29 \leqslant M_x = M_y \leqslant 99$, and $M_z = 3$. Since the mean fields are only a function of (x, y), more grid cells are used in the x–y plane. The particle Stokes number is set to 0.8. The maximum inflow volume fraction (cf. Eq. (41)) $\alpha_{p,\max} = 0.01$, and the minimum inflow volume fraction $\alpha_{p,\min} = 0.001$, which are typical values encountered in the LE simulations of dilute particle-laden flows.

As in the lid-driven cavity problem, two TLE simulation approaches are investigated. In TLE1 real particles are used. Particles are injected at $x = 0$ by defining an inlet volume $\mathscr{V}^{\mathrm{in}} = U_0 \Delta t \Delta y \Delta z$, such that the number of real particles introduced at each time step in cth cell that adjoins the boundary at $x = 0$ is given by

$$N_p^{\mathrm{in}}(y_c) = \left\lceil \frac{\alpha_p(x = 0, y_c)\mathscr{V}^{\mathrm{in}}}{\mathscr{V}_p} \right\rceil, \tag{45}$$

where $\lceil \cdot \rceil$ is the nearest greater integer operator, and y_c is the cell center coordinate. The fractional loss in actual injected volume fraction due to the greatest integer operation is saved and added to the N_p^{in} computation in the next time step. These $N_p^{\mathrm{in}}(y_c)$ particles are uniformly distributed in the volume $\mathscr{V}^{\mathrm{in}}$.

In TLE2, computational particles with equal and non-evolving statistical weights are used. The inflow of the real particles is indirectly implemented by a uniform inflow of computational particles, and the weight distribution of the injected computational particles mimics the inflow volume fraction. The number of computational particles N_c^{in} in the cth cell is computed as

$$N_c^{\mathrm{in}}(y_c) = \lceil n_c^{\mathrm{in}} \mathscr{V}^{\mathrm{in}} \rceil, \tag{46}$$

where $\lceil \cdot \rceil$ is the nearest greater integer operator, $\mathscr{V}^{\mathrm{in}}$ is the inlet volume defined earlier, $n_c{}^{\mathrm{in}} = N_{\mathrm{pc}}^{\mathrm{in}} / \mathscr{V}_m$ is the inflow number density of the computational particles, and $N_{\mathrm{pc}}^{\mathrm{in}}$ is the user specified parameter that determines the numerical resolution of TLE2 simulation. The statistical weight $W^{(i)}$ for each injected computational particle is

$$W^{(i)} = \frac{N_p^{\mathrm{in}}}{N_c^{\mathrm{in}}}, \quad i = 1, \ldots, N_c^{\mathrm{in}}. \tag{47}$$

For the ILE simulation, the computational particles are injected at $x = 0$ as in the TLE2 case. However, during the simulation, their weights evolve as a result of the particle number density control algorithm.

In order to meaningfully compare the accuracy of TLE2 and ILE it is necessary to maintain the same numerical resolution in both simulations. The number of computational particles per cell varies throughout the domain in both TLE2 and ILE, as does the total number of computational particles contained inside the region bounded by the bounding streamline (Eq. (42)). Furthermore, because the number of computational particles in each cell is a random variable (differs with each realization), it makes sense to only ensure that the average (or other statistics, such as min/max) number of computational particles per cell is the same in TLE2 and ILE. However, it is difficult to maintain exactly the same numerical resolution, even in terms of average number of computational particles in each cell, because of the nature of the simulation methods. The parameter that controls the computational particle distribution in TLE2 is $N_{\mathrm{pc}}^{\mathrm{in}}$, and for TLE, it is $N_{\mathrm{pc}}^{\mathrm{T}}$. Through trial and error, we have developed empirical rules that give approximately the same average number of computational particles per cell inside the bounding streamline for TLE2 and ILE simulations as 24 and 28, respectively. Additionally, for the finest grid used ($100 \times 100 \times 3$), the total number of computational particles inside the bounding streamline for TLE2 and ILE is equal to 690,000 and 747,282, respectively. These values are obtained with $N_{\mathrm{pc}}^{\mathrm{in}} = 23$ and $N_{\mathrm{pc}}^{\mathrm{T}} = 20$. In this way, a comparable numerical resolution is maintained between TLE2 and ILE simulations.

7. Results

We have calculated the mean interphase momentum transfer term using all the interpolation schemes (LPI-2, LPI-4, PCA, and TSE). However, only one set of representative contour plots of $\{\mathbf{F}^{\mathrm{fp}}\}$ and its relative error obtained using LPI-2 are reported here. Fig. 6a, c, e show, respectively, the contour plots of $\{F_y^{\mathrm{fp}}\} / \left\langle F_y^{\mathrm{fp}} \right\rangle_{\max}$ from TLE1, TLE2, and ILE simulations on the coarsest grid ($30 \times 30 \times 3$). For this grid resolution, all the three simulation methods yield nearly identical estimates. However, the contour plots for the finest grid ($100 \times 100 \times 3$) for TLE1 (Fig. 6(b)), TLE2 (Fig. 6(d)) and ILE (Fig. 6(f)) simulations, clearly show the worsening of estimates for the TLE1 although TLE2 and ILE do not give very different estimates.

Fig. 7(a), (c) and (e) show, respectively, the contour plots of relative error $\left(\left| \frac{\langle F_y^{\mathrm{fp}} \rangle - \{F_y^{\mathrm{fp}}\}}{\langle F_y^{\mathrm{fp}} \rangle} \right| \right)$ from TLE1, TLE2, and ILE simulations on the coarsest grid. The relative error for the finest grid resolution is shown in Fig. 7(b)(TLE1), Fig. 7(d) (TLE2), and 7(f) (ILE). For both resolutions, TLE1 gives the maximum error, ILE gives the minimum error, while errors incurred by TLE2 lie in the middle. The highest error in TLE1 simulation is due to the fewer number of particles per cell on progressively refined grids. The lower number of particles per cell on finer grids results in increased statistical error. This error is highest in the regions of low number density. On the $30 \times 30 \times 3$ grid (Fig. 7(a)) the relative error is nearly uniform over the entire domain. However, on the $100 \times 100 \times 3$ grid (Fig. 7(b)), the relative error becomes more than 100%, with the highest error observed in regions of low number density ($0.5 < y < 1.0$ and $0.0 < x < 1.0$).

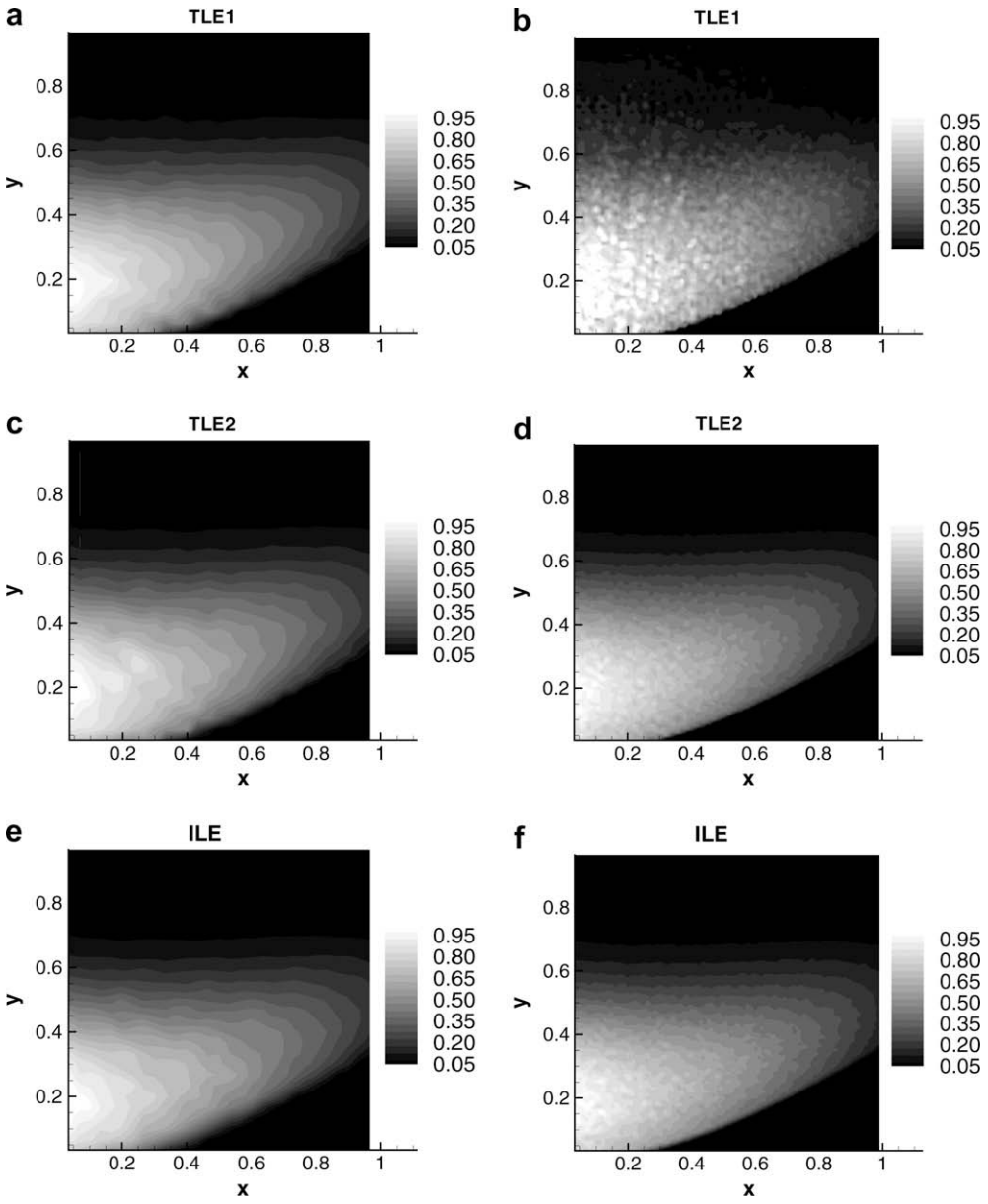

Fig. 6. (a, c, and e) Contour plots of the scaled mean interphase momentum transfer term $\{F_y^{fp}\}/\langle F_y^{fp}\rangle_{max}$ obtained from TLE1, TLE2, and ILE simulations of the test problem on a $30 \times 30 \times 3$ grid. (b, d, and f) Contour plots of the scaled mean interphase momentum transfer term obtained from TLE1, TLE2, and ILE simulations on a $100 \times 100 \times 3$ grid. Interpolation scheme used is LPI-2 with improved estimator.

It is interesting to note that for this test problem the TLE2 simulation, although less accurate than ILE simulation, provides reasonable estimates for the mean interphase momentum transfer term. This is because the fluid velocity field in the test problem has zero vorticity, and hence the particles do not preferentially concentrate. For this particular test problem, the computational particles in the TLE2 simulation maintain an acceptable particle number density even in the regions of low physical volume fraction. Therefore, the test problem does not result in highly non-uniform spatial distribution of particles that was encountered earlier in the lid-driven flow. As a result, the estimates from the TLE2 simulation do not worsen as drastically with grid refinement as in the more realistic lid-driven cavity flow. The test results show that the particle number density control algorithm yields highly accurate results that capture the flow physics.

We now use the test problem to investigate effect of the estimator on LPI-2, LPI-4, and PCA interpolation schemes. Fig. 8 compares the convergence characteristics of root mean square relative error for different interpolation schemes (LPI-2, LPI-4, and PCA) using the conventional estimator. Fig. 9 shows the same convergence characteristics but with estimates obtained from the improved estimator. Since the TSE is always implemented with improved estimator, its convergence characteristics are shown only for the improved estimator case. In both the figures, lines are simple fit to the data. For the TLE1 simulation method, the root mean square relative errors for all the interpolation schemes show that neither choice of estimator yields numerically converged results. Regardless of the choice of estimator in TLE1, the errors first decrease and then increase with grid refinement. On the other hand, the rms relative errors from TLE2 (dashed lines) and ILE (dashed dot lines) simulations show a monotonic decrease for both estimators, with ILE being the more accurate. From these observations, we conclude that ILE along with the improved estimator will result in numerically converged and accurate LE simulations.

The rate of convergence of the rms relative errors using the conventional and the improved estimators is obtained by performing

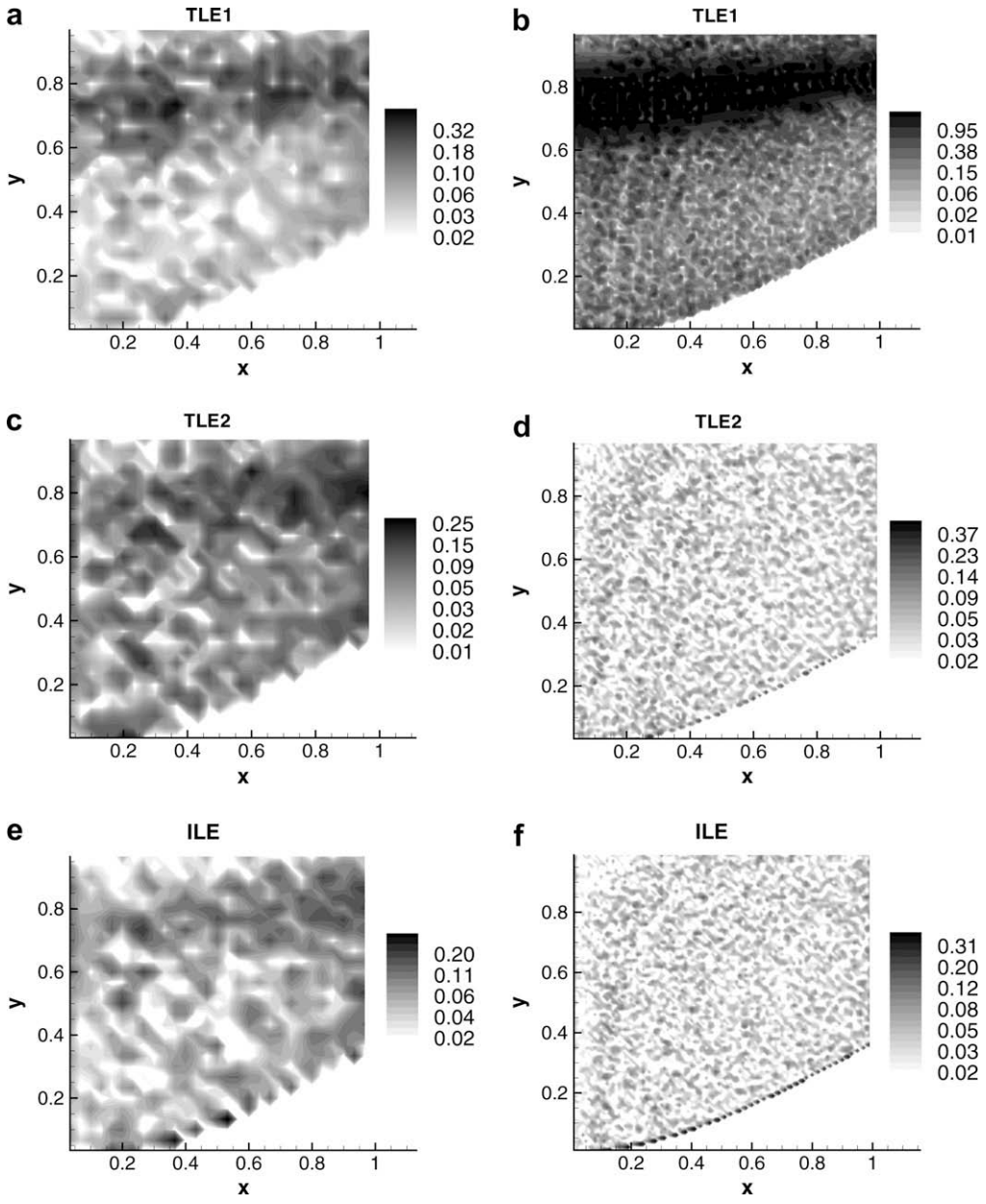

Fig. 7. (a, c, and e) Contour plots of relative error $\left(\left|\frac{\langle F_y^{fp}\rangle - \{F_y^{fp}\}}{\langle F_y^{fp}\rangle}\right|\right)$ obtained from TLE1, TLE2, and ILE simulations on a $30 \times 30 \times 3$ grid. (b, d, and f) Contour plots of relative error obtained from TLE1, TLE2, and ILE simulations on a $100 \times 100 \times 3$ grid. The LPI-2 interpolation scheme is used with improved estimator.

linear least-squares fit to the data in Figs. 8 and 9. The convergence rates are summarized in Table 1 for all the interpolation schemes. The convergence rate of the rms relative errors is not reported for TLE1 because it shows no signs of convergence. From Table 1, it is observed that the improved estimator consistently gives higher rates of convergence for LPI-2, LPI-4, and PCA interpolation schemes as compared to the conventional estimator. The first step in the TSE algorithm is identical to the improved estimator (Dreeben and Pope, 1992; Subramaniam and Haworth, 2000; Garg et al., 2007). Therefore, for TSE, the rate of convergence of the rms relative error is independent of the estimator used.

8. Discussion

Particle-based methods have been used extensively in many fields other than two-phase flows. For example, in single-phase turbulent reactive flows, the so called "hybrid particle/finite-volume PDF method" is nowadays commonly used. In this approach, the flow is solved using a standard finite volume method. However, in order to avoid the use of closures for chemical reaction terms, a stochastic differential equation is used to solve for species evolution. The stochastic differential equations, solved using a Monte–Carlo approach, result in finite number of stochastic particles that are used for species transport. For constant density flows, these stochastic particles are always uniformly distributed resulting in spatially uniform distribution of statistical error. For variable density flows, however, the number density of the stochastic particles, if not corrected, can become highly non-uniform. In order to avoid spatially non-uniform distribution of statistical error, particle number density control algorithms have often been employed in the simulations of turbulent reactive flows (Pope, 1985; Haworth and Tahry, 1991; Subramaniam and Haworth, 2000; Jaberi et al., 1999; Raman et al., 2005). LE simulations using real particles (or computational particles with constant statistical weight) also suffer from spatially non-uniform distribution of statistical error as

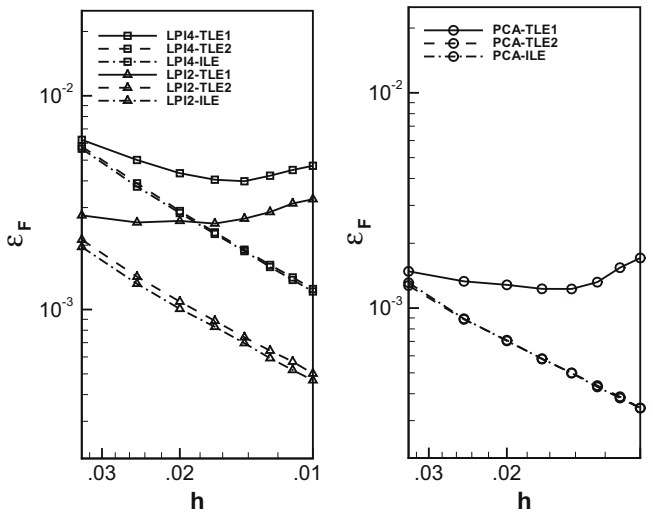

Fig. 8. Convergence characteristics of the root mean square relative error with grid spacing h for TLE1 (solid), TLE2 (dash), and ILE (dash–dot) simulations of the test problem. Conventional estimator is used. Lines are simple fit to the symbols. □, LPI-4; △, LPI-2; ○, PCA.

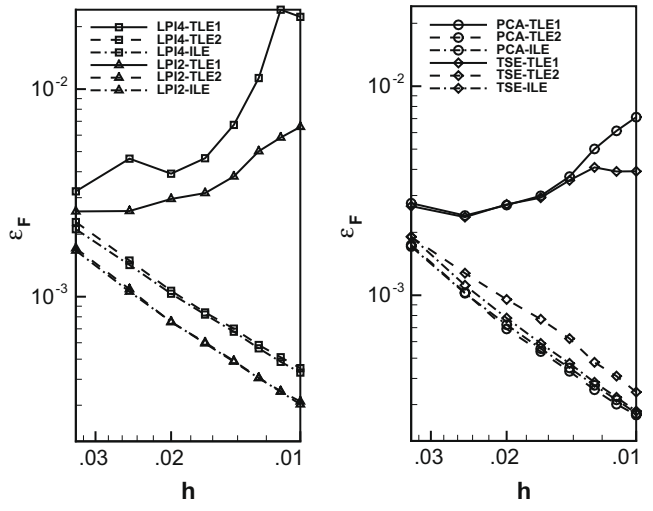

Fig. 9. Convergence characteristics of the root mean square relative error with grid spacing h for TLE1 (solid), TLE2 (dash), and ILE (dash–dot) simulations of the test problem. Improved estimator is used. Lines are simple fit to the symbols. □, LPI-4; △, LPI-2; ○, PCA, ◇, TSE.

Table 1
Comparison of relative root mean square error's convergence rate between conventional and improved estimator for all the estimation schemes in TLE2 and ILE simulations.

	Conventional estimator		Improved estimator	
	TLE2	ILE	TLE2	ILE
LPI-4	1.27	1.27	1.34	1.32
LPI-2	1.18	1.18	1.41	1.40
PCA	1.07	1.09	1.54	1.50
TSE	—	—	1.42	1.58

the particle number density can go to zero in some regions of the flow. Therefore, a particle number density control algorithm, like the one used in turbulent reactive flows, becomes imperative in order to ensure accurate LE simulations in multiphase flows. The test cases considered in this study demonstrate the accuracy and convergence of the particle number density control algorithm incorporated into the ILE method, but they are relatively simple in that the regions of the flow devoid of particles do not change drastically in time. While we do not anticipate any special difficulties with simulating such flows, they may be suitable test problems for future study.

Although the particle number control algorithm ensures that the statistical error is uniformly distributed over the entire flow domain, the accuracy of numerically estimated mean interphase momentum transfer term is only as good as the estimator used. A simple modification to the conventionally used estimation procedure for mean interphase momentum transfer term gives more accurate estimates along with a higher rate of convergence for all simulation methods. Although the improved estimator gives a big improvement over the conventional estimator, some caution should be exercised in choosing the interpolation scheme when using improved estimator. This is due to the difference in basis function definitions for each scheme. Interpolation schemes like LPI-2, and PCA have strictly positive basis function values, therefore, both the numerator and the denominator in Eq. (30) always scale well, even in the limit of low number of particles per cell. Basis function for LPI-4, on the other hand, can become both positive and negative. As a result, in the limit of low N_{pc}, the denominator in Eq. (30) may acquire a very small value that does not scale well with the numerator, resulting in poor estimates. Therefore, the use of LPI-4 is not recommended with the improved estimator.

TLE simulations suffer from increased statistical error with grid refinement, resulting in their failure to yield numerically converged estimates. The limitations of TLE simulations can be overcome by ensuring that the statistical error remains constant on progressively refined grids, and as well as is spatially uniformly distributed. In typical LE simulations, including ours, estimates for the mean interphase momentum transfer term are formed using a kernel whose support or bandwidth scales with grid size. These are generally referred to as the grid-cell based estimators and, as observed in this study, they suffer from increased statistical error with grid refinement. If estimation kernel's bandwidth remains constant, then the statistical error will also remain constant with grid refinement. Such estimators are referred to as the fixed-bandwidth or grid-free estimators. For example, if the bandwidth of the top hat kernel in Fig. 2 is kept fixed at h, then at any spatial location the number of samples used to form the mean field estimates is approximately the same for both the coarse (solid vertical lines) grid and the fine grid (dashed vertical lines). This ensures constant statistical error on both coarse and fine grids. However, even with fixed-bandwidth estimators, the statistical error can be spatially non-uniform in flows with preferential concentration. Also fixed-bandwidth estimators do not show improved accuracy with grid refinement because the discretization error[4] in the estimate scales as a power of the bandwidth, independent of the grid-size. Therefore, although fixed-bandwidth estimators are superior to grid-cell based estimators and aid in overall stability of LE simulations, they do not solve the problem of spatially non-uniform distribution of particles. In this context, our ILE simulation method fulfills both the desired objectives: (a) near-constant statistical error and decreasing discretization error with grid refinement, and (b) spatially near-uniform distribution of statistical error.

9. Conclusions

In LE simulations of two-phase flows the spatial distribution of particles can become highly non-uniform due to preferential concentration, if the Stokes number is in the appropriate range. Such

[4] The discretization error for fixed bandwidth kernel (Dreeben and Pope, 1992) scales as $\mathcal{O}(h^p)$, where p depends on the estimation scheme.

situations are frequently encountered in two-phase flows. Simulations of a particle-laden lid-driven cavity flow show that traditional LE simulations are not numerically convergent. An improved LE simulation approach is developed that maintains near-uniform computational particle number density, resulting in a numerically convergent solution to the particle-laden lid-driven cavity problem. In order to establish the accuracy of the ILE method, a novel two-phase flow test problem that admits an analytical solution for the mean interphase momentum transfer term is devised. This test reveals that the ILE method yields accurate solutions also. Numerical tests reveal that an improved estimator yields very accurate estimates compared to the conventional estimator that is currently used in LE simulations. Therefore, the combination of ILE with the improved estimator yields numerically convergent and accurate results for two-phase flows.

Acknowledgments

S.S. and R.G. thank Djamel Lakehal at ASCOMP GmbH for making the particle tracking code available.

References

Abraham, J., 1997. What is adequate resolution in the numerical computations of transient jets? SAE Paper 970051.

Are, S., Hou, S., Schmidt, S.P., 2005. Second-order spatial accuracy in Lagrangian–Eulerian spray calculations. Numer. Heat Transfer B 48, 25–44.

Boivin, M., Simonin, O., Squires, K.D., 1998. Direct numerical simulation of turbulence modulation by particles in isotropic turbulence. J. Fluid Mech. 375, 235–263.

Dreeben, T.D., Pope, S.B., 1992. Nonparametric estimation of mean fields with application to particle methods for turbulent flows. Tech. Rep. FDA 92-13, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853.

Drew, D.A., Passman, S.L., 1998. Theory of Multicomponent Fluids. Applied Mathematical Sciences. Springer, New York.

Elghobashi, S.E., Truesdell, G.C., 1993. On the two-way interaction between homogeneous turbulence and dispersed solid particles I: turbulence modification. Phys. Fluids A 5, 1790–1801.

Fan, R., Fox, R.O., 2008. Segregation in polydisperse fluidized beds: validation of a multi-fluid model. Chem. Eng. Sci. 63 (1), 272–285. http://www.dx.doi.org/10.1016/j.ces.2007.09.038.

Fan, R., Marchisio, D.L., Fox, R.O., 2004. Application of the direct quadrature method of moments to polydisperse gas–solid fluidized beds. Powder Technol. 139, 7–20.

Fox, R.O., 2003. Computational Models for Turbulent Reacting Flows. Cambridge University Press.

Garg, R., Narayanan, C., Lakehal, D., Subramaniam, S., 2007. Accurate numerical estimation of interphase momentum transfer in Lagrangian–Eulerian simulations of dispersed two-phase flows, doi:10.1016/j.ijmultiphaseflow.2007.06.002.

Haworth, D.C., Tahry, S.H.E., 1991. Probability density function approach for multidimensional turbulent flow calculations in reciprocating engines. AIAA J. 29, 208–218.

Jaberi, F.A., Colucci, P.J., James, S., Givi, P., Pope, S.B., 1999. Filtered mass density function for large-eddy simulation of turbulent reacting flows. J. Fluid Mech. 401, 85–121.

Kannenberg, K., Boyd, I., 2000. Strategies for efficient particle resolution in the direct simulation Monte Carlo method. J. Comput. Phys. 157, 727–745.

Kim, J., Moin, P., 1985. Application of a fractional-step method to incompressible Navier–Stokes equations. J. Comput. Phys. 59, 308–323.

Moreau, M., Fede, P., Simonin, O., Villedieu, P., 2003. Monte carlo simulation of colliding particles suspended in gas–solid homogeneous turbulent shear flows, vol. 2A, pp. 491–500.

Narayanan, C., Lakehal, D., Yadigaroglu, G., 2002. Linear stability analysis of particle-laden mixing layers using particle tracking. Powder Technol. 125, 122–130.

Patankar, N.A., Joseph, D.D., 2001. Lagrangian numerical simulation of particulate flows. Int. J. Multiphase Flow 27, 1685–1706.

Pope, S.B., 1985. PDF methods for turbulent reactive flows. Prog. Energy Combust. Sci. 11, 119–192.

Raman, V., Pitsch, H., Fox, R.O., 2005. Hybrid large-eddy simulation/Lagrangian filtered-density-function approach for simulating turbulent combustion. Combust. Flame 143, 56–78.

Snider, D.M., O'Rourke, P.J., Andrews, M.J., 1998. Sediment flow in inclined vessels calculated using a multiphase particle-in-cell model for dense particulate flows. Intl. J. Multiphase Flow 24, 1359–1382.

Squires, K.D., Eaton, J.K., 1990. Particle response and turbulence modification in isotropic turbulence. Phys. Fluids A 2, 1191–1203.

Subramaniam, S., 2000. Statistical representation of a spray as a point process. Phys. Fluids 12, 2413–2431.

Subramaniam, S., 2001. Statistical modeling of sprays using the droplet distribution function. Phys. Fluids 13, 624–642.

Subramaniam, S., Haworth, D.C., 2000. A pdf method for turbulent mixing and combustion on three-dimensional unstructured deforming meshes. J. Engine Res. 1, 171–190.

Sundaram, S., Collins, L.R., 1996. Numerical considerations in simulating a turbulent suspension of finite-volume particles. J. Comp. Phys. 124, 337–350.

Williams, F.A., 1958. Spray combustion and atomization. Phys. Fluids 1, 541–545.