

2009

# Modeling and Simulation of two-phase flows

Rahul Garg  
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/etd>

 Part of the [Mechanical Engineering Commons](#)

## Recommended Citation

Garg, Rahul, "Modeling and Simulation of two-phase flows" (2009). *Graduate Theses and Dissertations*. 10657.  
<https://lib.dr.iastate.edu/etd/10657>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

**Modeling and simulation of two-phase flows**

by

Rahul Garg

A dissertation submitted to the graduate faculty  
in partial fulfillment of the requirements for the degree of  
**DOCTOR OF PHILOSOPHY**

Major: Mechanical Engineering

Program of Study Committee:  
Shankar Subramaniam, Major Professor  
Rodney O. Fox  
Theodore J. Heindel  
Michael G. Olsen  
Richard H. Pletcher  
Scott W. Hansen

Iowa State University

Ames, Iowa

2009

Copyright © Rahul Garg, 2009. All rights reserved.

## Dedication

Dedicated to my parents for their lifetime of support and encouragement.

## TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	viii
<b>LIST OF FIGURES</b> . . . . .	ix
<b>ABSTRACT</b> . . . . .	xxiii
<b>CHAPTER 1. INTRODUCTION</b> . . . . .	1
1.1 Background . . . . .	1
1.2 Research Objectives . . . . .	9
1.2.1 Modeling interphase momentum transfer term using ‘true’ DNS . . . . .	9
1.2.2 Modeling mean interphase heat and mass transfer term using ‘true’ DNS	13
1.2.3 Numerical Accuracy and Convergence Characteristics of LE simulations	15
1.2.4 Development of an open source DEM code for simulations of granular and gas–particle systems . . . . .	18
1.3 Thesis outline . . . . .	18
<b>CHAPTER 2. DIRECT NUMERICAL SIMULATION OF GAS-SOLIDS FLOW BASED ON THE IMMERSED BOUNDARY METHOD</b> . . . . .	21
2.1 Introduction . . . . .	21
2.1.1 Transport of the particle distribution function . . . . .	22
2.1.2 Homogeneous suspension flow . . . . .	23
2.1.3 Steady flow past homogeneous assemblies of fixed particles . . . . .	23
2.2 Governing equations . . . . .	24
2.3 The immersed boundary method . . . . .	27
2.3.1 Solution Approach . . . . .	29

2.3.2	Estimation of mean pressure gradient $\langle \mathbf{g}_{\text{IBM}} \rangle_V$ . . . . .	33
2.4	Simulation Methodology . . . . .	36
2.4.1	Numerical parameters . . . . .	37
2.4.2	Estimation of mean drag from simulations . . . . .	38
2.4.3	Numerical Convergence . . . . .	40
2.5	Validation Tests . . . . .	44
2.5.1	Isolated Sphere . . . . .	44
2.5.2	Stokes Flow . . . . .	45
2.6	Ordered Arrays . . . . .	47
2.7	Random Arrays . . . . .	49
2.7.1	Dilute Arrays . . . . .	50
2.7.2	Moderately Dilute to Dense Arrays . . . . .	51
2.7.3	Numerical Parameters and Resolution . . . . .	53
2.8	Summary . . . . .	55
2.9	Assessment of IBM for drag law formulation . . . . .	55
2.10	Conclusion . . . . .	58

### **CHAPTER 3. SCALAR TRANSPORT AND HEAT TRANSFER PAST ORDERED AND RANDOM ARRAYS OF MONODISPERSE SPHERES IN STOKES FLOW . . . . .**

3.1	Introduction . . . . .	61
3.2	Governing Equations . . . . .	68
3.2.1	Implications of simulation assumptions . . . . .	71
3.3	Solution Approach . . . . .	73
3.3.1	Specification of the sink/source term $S_{\phi, \text{IBM}}$ . . . . .	75
3.4	Simulation Methodology . . . . .	77
3.4.1	Estimation of Nusselt number from simulations . . . . .	77
3.5	Numerical convergence . . . . .	78
3.5.1	Numerical method validation . . . . .	82

3.6	Results . . . . .	84
3.6.1	Stokes flow . . . . .	84
3.6.2	Budget of the scalar transport equation . . . . .	87
3.6.3	A correlation for average Stokes Nusselt number $Nu_0$ . . . . .	90
3.6.4	Effects of inter-particle distance on the local Nusselt number and local viscous drag . . . . .	90
3.6.5	Effects of hard-core distance $h_c$ on average Stokes Nusselt number $Nu_0$ in random arrays . . . . .	103
3.7	Discussion and Conclusions . . . . .	108

<b>CHAPTER 4. SCALAR TRANSPORT AND HEAT TRANSFER PAST ORDERED AND RANDOM ARRAYS OF MONODISPERSE SPHERES IN FLOWS WITH FINITE FLUID INERTIA . . . . .</b>		110
4.1	Results . . . . .	111
4.1.1	Ordered arrays . . . . .	111
4.1.2	Random arrays . . . . .	118
4.1.3	Budget of the scalar transport equation . . . . .	119
4.2	A correlation for average Nusselt number $Nu$ . . . . .	122
4.3	Discussion and conclusions . . . . .	122

<b>CHAPTER 5. ACCURATE NUMERICAL ESTIMATION OF INTERPHASE MOMENTUM TRANSFER IN LAGRANGIAN-EULERIAN SIMULA- TIONS OF DISPERSED TWO-PHASE FLOWS . . . . .</b>		125
5.1	Introduction . . . . .	126
5.1.1	Physical system . . . . .	126
5.1.2	Computational representation . . . . .	129
5.1.3	Problem Statement . . . . .	130
5.1.4	Review of existing schemes . . . . .	131
5.2	Test Problem . . . . .	138
5.3	Numerical Analysis . . . . .	141

5.3.1	Error Decomposition . . . . .	143
5.4	Results . . . . .	145
5.4.1	Test 1: Baseline test case . . . . .	145
5.4.2	Test 2: Effect of nonzero particle velocity variance . . . . .	155
5.4.3	Test 3: Variation of particle position distribution . . . . .	157
5.4.4	Test 4: Variation of fluid velocity field . . . . .	160
5.5	Comparison with representative LE numerical parameters . . . . .	162
5.6	Conclusions . . . . .	162
<b>CHAPTER 6. A NUMERICALLY CONVERGENT LAGRANGIAN–EULERIAN</b>		
<b>SIMULATION METHOD FOR DISPERSED TWO–PHASE FLOWS . . . . .</b>		
6.1	Introduction . . . . .	167
6.2	Governing Equations . . . . .	172
6.3	Statistical Description of Dispersed Phase: Computational Particles . . . . .	175
6.3.1	Traditional LE Simulation (TLE): Equally– weighted particles . . . . .	176
6.3.2	Improved LE Simulation (ILE): Unequal and Evolving weights . . . . .	177
6.4	Numerical Estimation of Mean Interphase Momentum Transfer Term . . . . .	181
6.5	Lid–Driven Cavity Flow Problem . . . . .	186
6.6	Test Problem . . . . .	189
6.6.1	Computation setup . . . . .	193
6.7	Results . . . . .	196
6.8	Discussion . . . . .	201
6.9	Conclusions . . . . .	203
<b>CHAPTER 7. DEVELOPMENT OF AN OPEN SOURCE DEM CODE</b>		
<b>FOR SIMULATIONS OF GRANULAR AND GAS–PARTICLE SYS-</b>		
<b>TEMS . . . . .</b>		
7.1	Theory: Introduction and Background . . . . .	207
7.2	Theory and Numerical Implementation . . . . .	207
7.2.1	Gas-phase Computations . . . . .	208

7.2.2	Discrete Element Method: DEM . . . . .	208
7.3	Neighbour Search Algorithm . . . . .	216
7.4	DEM Verification Tests . . . . .	218
7.4.1	Freely Falling Particle . . . . .	218
7.4.2	Ball Slipping on a Rough Surface . . . . .	219
7.5	Future Work . . . . .	221
<b>CHAPTER 8. DISCUSSION AND CONCLUSIONS . . . . .</b>		<b>223</b>
8.1	Conclusions from modeling study of fluid–particle interaction term . . . . .	223
8.2	Conclusions from study of numerical accuracy and convergence characteristics of LE simulations . . . . .	226
8.3	Implications of KTGF closures for particle–particle interaction term in EE sim- ulations . . . . .	229
<b>APPENDIX A. RANDOM CONFIGURATIONS INITIALIZATION . . . . .</b>		<b>233</b>
<b>APPENDIX B. DETAILS OF INTERPOLATION SCHEMES . . . . .</b>		<b>240</b>
B.1	Linear Interpolation (LPI-2) . . . . .	242
B.2	Lagrange Polynomial Interpolation (LPI) . . . . .	242
B.3	Piecewise Cubic Approximation (PCA) . . . . .	242
B.4	Two Stage Estimation Algorithm . . . . .	243
<b>APPENDIX C. DETAILS OF ERROR DECOMPOSITION . . . . .</b>		<b>246</b>
C.1	Statistical Error . . . . .	246
<b>APPENDIX D. DETAILS OF DDF EQUATION DERIVATION . . . . .</b>		<b>248</b>
D.1	Evolution equation for $h$ . . . . .	248
D.2	Evolution equation for $\langle N_p(t) \rangle$ . . . . .	249
D.3	Number density evolution equation . . . . .	249
D.4	Evolution of the velocity conditional pdf . . . . .	250
<b>BIBLIOGRAPHY . . . . .</b>		<b>252</b>
<b>ACKNOWLEDGEMENTS . . . . .</b>		<b>267</b>

## LIST OF TABLES

Table 2.1	Comparison of the numerical resolutions used for random arrays in IBM simulations with the past LBM simulations of HKL and BVK. For each entry, first and second rows correspond, respectively, to the LBM simulations of HKL and BVK, and the third row corresponds to the current IBM simulations. For the IBM simulations, the numbers before and after “,” are, respectively, the resolutions for $Re \leq 100$ and $Re > 100$ .	54
Table 5.1	Representation of carrier flow and dispersed phase in different LE simulations: DNS <sup>(b)</sup> and LES <sup>(b)</sup> are denoted hybrid simulations. . . . .	128
Table 6.1	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. . . . .	200

## LIST OF FIGURES

Figure 1.1	A schematic showing a realization of two-phase flow and various simulation types that can be used to solve for a realization ('true' DNS, discrete element method DEM) or the resulting averaged equations (LE, EE, and QMOM). The simulation types in <i>bold italics</i> are the ones considered in this work. . . . .	4
Figure 1.2	Schematic showing a cloud of gas-phase along with dispersed phase solid particles (solid-circles). In simulation solving for averaged equations (such as LE and EE), modeling of gas-solid interactions, such as interphase momentum transfer $I_{gp}$ , interphase heat and mass transfer $\gamma_{gp}$ , etc., and particle-particle interaction force $I_{pp}$ are required. . . .	5
Figure 1.3	Schematic showing the modeling requirements (shown inside the dashed box) for two-phase flow simulations of averaged equations (LE, EE, and QMOM), and also the sources of models for fluid-particle and particle-particle interaction terms. . . . .	6
Figure 2.1	A simple schematic of the physical domain . . . . .	25
Figure 2.2	Schematic of the computational domain along with particles in IBM simulations . . . . .	29
Figure 2.3	Convergence characteristics of drag force due to fluctuating pressure gradient (open symbols) and viscous stresses (filled symbols) for FCC arrays with grid resolution $D_m$ for two CFL values of 0.2 (squares) and 0.05 (triangles). $Re = 40$ and volume fraction $\varepsilon_s$ is equal to 0.2 in (a) and 0.4 in (b). . . . .	40

- Figure 2.4 Convergence characteristics for random arrays at  $Re = 20$  of the normalized force with box length to particle diameter ratio  $L/D$  for four different values of  $D_m$  equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Solid volume fraction is equal to 0.3 in (a) and 0.4 in (b). Drag values have been averaged over 5 multiple independent simulations. Not all combinations of  $D_m$  and  $L/D$  are shown because with a serial code some combinations exceeded computational memory requirements. . . . . 42
- Figure 2.5 Normalized drag force  $F$  in a simple cubic array ( $\varepsilon_s = 4.0 \times 10^{-4}$ ) as a function of Reynolds number and angle  $\theta$  between the mean flow and the  $x$ - axis in the  $(x, y)$  plane. The symbols are from the IBM simulations:  $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ ). The solid line is the drag correlation for an isolated sphere in unbounded medium (Schiller and Naumann, 1933). . . . . 44
- Figure 2.6 Comparison of the normalized drag force  $F$  as a function of the solid volume fraction  $\varepsilon_s$  in in Stokes flow past SC and FCC arrays from IBM simulations (open symbols) with the results of Zick and Homsy (filled symbols). . . . . 46
- Figure 2.7 Comparison of the normalized drag force  $F$  for SC arrangement obtained from IBM (open symbols) simulations with the LBM simulations (filled symbols) of HKL. The solid line is the Schiller and Naumann drag law for a single particle in an unbounded medium. The flow is directed along the  $x$ - axis. . . . . 47
- Figure 2.8 Comparison of the normalized drag force  $F$  for FCC arrangement obtained from IBM (open symbols) simulations with the LBM simulations (filled symbols) of HKL. The solid line is the Schiller and Naumann drag law for a single particle in an unbounded medium. The flow is directed along the  $x$ - axis. . . . . 48

Figure 2.9	Normalized drag force $F$ Vs Reynolds number for random configuration at solid volume fraction equal to 0.01. Symbols are the IBM simulations, where for squares the flow is directed along the $x$ - axis, and for triangles the flow is directed at an angle of $\pi/16$ in the $x - y$ plane. . . . .	50
Figure 2.10	Comparison of the normalized drag force $F$ for random arrays at volume fractions equal to 0.1 and 0.2 from IBM simulations (open symbols) with the monodisperse drag laws of HKL and BVK. . . . .	52
Figure 2.11	Comparison of the normalized drag force $F$ for random arrays at volume fractions equal to 0.3 and 0.4 from IBM simulations (open symbols) with the monodisperse drag laws of HKL and BVK. . . . .	52
Figure 3.1	Schematic of various heat transfer mechanisms in suspensions. . . . .	63
Figure 3.2	Convergence characteristics of Nusselt number with grid resolution $D_m$ for FCC arrays for two different solid volume fractions equal to 0.2 (triangles) and 0.4 (squares) at (a) $Re = 0.01$ and (b) $Re = 40$ . Open and filled symbols are for CFL values of 0.2 and 0.05, respectively. . .	79
Figure 3.3	Convergence characteristics for random arrays at $Re = 20$ of the Nusselt number with box length to particle diameter ratio $L/D$ for four different values of $D_m$ equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Solid volume fraction is equal to 0.3 in (a) and 0.4 in (b). Nusselt number values have been averaged over 5 multiple independent simulations. Not all combinations of $D_m$ and $L/D$ are shown because with a serial code some combinations exceeded computational memory requirements. . . . .	80

Figure 3.4 (a) Normalized drag force  $F$  and (b) Nusselt number in a simple cubic array ( $\varepsilon_s = 4.0E - 04$ ) as a function of Reynolds number and angle  $\theta$  between the mean flow and the  $x$ - axis in the  $(x, y)$  plane. The symbols are from the IBM simulations:  $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ ). Lines are a simple fit to the data. In (a) and (b), the solids lines are the single-sphere correlations for normalized drag force (Schiller and Naumann (1933)) and Nusselt number (Eq. 3.32), respectively. Also in (b), the dashed line is the Nusselt number from Gunn’s correlation at infinite dilution ( $\varepsilon_s = 0$ ). . . . . 83

Figure 3.5 Comparison of (a) the average Stokes Nusselt number  $Nu_0$  and (b) the scaled (by Stokes drag) Stokes viscous drag force  $F_{vis,0}$  versus the solid volume fraction  $\varepsilon_s$  between SC, FCC, and random arrangements in the Stokes flow regime. . . . . 85

Figure 3.6 Comparison of relative magnitudes of the “ensemble averaged” local fluid-phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_v$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_v$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_v$  (dash-dot line) along the  $x$ - axis obtained for the normalized convection  $\widehat{C}_\phi$ , axial diffusion  $\widehat{D}_{\phi,\parallel}$  and perpendicular diffusion terms  $\widehat{D}_{\phi,\perp}$ , respectively, in the normalized fluid-phase scalar transport equation 3.33. Ensemble average has been performed over 5 independent realizations, and the scatter between the realizations is shown by the one-sided error bars. . . . . 89

Figure 3.7 Schematic of the computational box along with solid particles. The contour plots of the non-dimensional scalar flux magnitude in the next few figures are shown in  $x$ - $y$  plane which lies midway along the  $z$ - axis. 91

Figure 3.8 Contour plots of the non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for SC and FCC arrays at different volume fractions in the Stokes flow regime. The flow is directed along the  $x$ - (from left to right in the figures) axis and the contour plots are shown in the  $x$ - $y$  plane lying midway along the  $z$ - axis as shown by the schematic in Fig. 3.7. . . . . . 93

Figure 3.9 Schematic of the spherical coordinate system used to define local Nusselt number and local viscous drag.  $\phi$  ( $0 \leq \phi \leq \pi$ ) is the polar angle and  $\theta$  ( $0 \leq \theta \leq 2\pi$ ) is the azimuthal angle. . . . . 94

Figure 3.10 (a), (c), and (e): Comparison of the local Nusselt number  $Nu^{loc}(\phi)$  (Eq. 3.36) along the polar angle  $\phi$  for SC, FCC, and random arrays (local Nusselt number averaged over all bodies), respectively. (b), (d), and (f): Comparison of the local Nusselt number  $Nu^{loc}(\theta)$  (Eq. 3.37) along the azimuthal angle  $\theta$  for SC, FCC, and random arrays, respectively. 96

Figure 3.11 (a) and (c): Comparison of the local viscous drag  $F_{vis}^{loc}(\phi)$  (Eq. 3.38) along the polar angle  $\phi$  for SC and random arrays, respectively. (b) and (d): Comparison of the local viscous drag  $F_{vis}^{loc}(\theta)$  along the azimuthal angle  $\theta$  for SC and random arrays, respectively. . . . . 99

Figure 3.12 Comparison of the local viscous drag  $F_{vis}^{loc}(\phi)$  (Eq. 3.38) for 10 randomly chosen particles out of a total of 95 particles from a realization of random array with solid volume fraction  $\varepsilon_s = 0.4$ . . . . . 101

Figure 3.13 (a) and (b): Comparison of the contour plots of non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for random arrays with solid volume fraction  $\varepsilon_s = 0.2$  for hard-core distance  $h_c$  values equal to  $D$  and  $1.35D$ , respectively. (c) and (d): Comparison of the local Nusselt number  $\text{Nu}^{\text{loc}}(\phi)$  along the polar angle  $\phi$  on 20 randomly chosen particles for the same solid volume fraction and hard-core distances. Since the contour plots have been shown on a x-y plane lying midway along the z- axis (see schematic in Fig. 3.7), only the projected area of particles is visible in this plane. . . . . 103

Figure 3.14 Comparison of the local average Nusselt number  $\text{Nu}^{\text{loc}}(\phi)$  (Eq. 3.36) along  $\phi$  obtained by averaging over all particles for different values of hard-core distance  $h_c$  in random arrays for (a)  $\varepsilon_s = 0.01$ , (b)  $\varepsilon_s = 0.1$ , (c)  $\varepsilon_s = 0.2$ , and (d)  $\varepsilon_s = 0.3$ . . . . . 105

Figure 3.15 Comparison of the average Stokes Nusselt number versus hard-core distance  $h_c$  in random arrays with SC and FCC arrays for solid volume fraction  $\varepsilon_s$  equal to 0.01 in (a) and 0.1 in (b). The slopes of the solid lines obtained by linear least-squares fit is indicated next to them. . . . . 106

Figure 3.16 The correction term due to finite hard-core distance  $\alpha_{\text{Nu}}^{h_c}$  (Eq. 3.42) in the Stokes flow versus solid volume fraction  $\varepsilon_s$ . The solid line obtained as a quadratic fit is  $17.7\varepsilon_s + 99.39\varepsilon_s^2$ . . . . . 107

Figure 4.1 Comparison of the average Nusselt number in SC arrays (a) versus the Reynolds number  $\text{Re}$  for different values of solid volume fraction, and (b) versus the solid volume fraction  $\varepsilon_s$  for different values of Reynolds number with Gunn's correlation (solid and dashed lines in both (a) and (b)) and Nusselt number correlation (dash-dotted line) for isolated particle (Eq. 3.32) in (a). The flow is directed along x-axis for all the above cases. . . . . 112

Figure 4.2 Comparison for different solid volume fractions in SC arrays of the local Nusselt number  $Nu^{loc}(\phi)$  (Eq. 3.36) along the polar angle  $\phi$  for Re equal to (a) 1, (b) 10, (c) 100, and (d) 300. . . . . 113

Figure 4.3 Contour plots of the non-dimensional scalar flux magnitude  $|\nabla\phi| D / |\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for SC arrays with  $\varepsilon_s = 0.3$  at (a) Re = 1, (b) Re = 10, (c) Re = 100, and (d) Re = 300. The flow is directed along  $x$ - axis and the contour plots are shown in the  $x$ - $y$  plane lying midway along the  $z$ - axis as shown by the schematic in Fig. 3.7. Few streamlines in (a) seem to be penetrating the particle surface. This is an artifact of the visualization software that draws streamlines based on interpolated values from the Eulerian grid, resulting in finite but negligible fluid velocity even at fluid-particle interface. 115

Figure 4.4 Comparison of the average Nusselt number in FCC arrays versus the Reynolds number Re for different values of solid volume fraction with Gunn’s correlation (solid and dashed lines) and Nusselt number correlation (dash-dotted line) for isolated particle (Eq. 3.32). The flow is directed along  $x$ -axis for all the above cases. . . . . 117

Figure 4.5 Comparison of the average Nusselt number in random arrays versus the Reynolds number Re for solid volume fractions equal to 0.1 and 0.2 from IBM simulations with Gunn’s correlation. . . . . 118

Figure 4.6 Comparison of the average Nusselt number in random arrays versus the Reynolds number Re for solid volume fractions equal to 0.3, 0.4, and 0.5 from IBM simulations with Gunn’s correlation. Due to the restriction posed by the serial solver, high numerical resolutions required for  $\varepsilon_s = 0.5$  and Re > 100 are not simulated. . . . . 120

Figure 4.7 Comparison of relative magnitudes of the “ensemble averaged” local fluid–phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_V$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_V$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_V$  (dash–dotted line) along the  $x$ - axis obtained for the normalized convection  $\widehat{C}_\phi$ , axial diffusion  $\widehat{D}_{\phi,\parallel}$  and perpendicular diffusion terms  $\widehat{D}_{\phi,\perp}$ , respectively. Ensemble average has been performed over 5 independent realizations, and the scatter between the realizations is shown by the one–sided error bars. . . . . 121

Figure 5.1 Sketch showing the PIC and PNN mean estimation schemes: (a) Mean estimation by PIC method in 2–D. Grid node 1 receives the full contribution from particles (shown as black spheres) located in cell area (shown by dotted lines) around it. (b) Mean estimation by PNN method in 2–D. For a particle (shown as a black sphere) in a two-dimensional cell, grid node 1 receives a fraction of the particle force which is proportional to the area of region 1 divided by the entire cell area. . . . . 134

Figure 5.2 (a) Basis functions for LPI-4. (b) Basis functions for PCA. In both figures, squares represent the fluid velocity at that grid node,  $X^{(k)}$  is the location of particle (shown by black sphere) located between nodes  $m$  and  $m + 1$ . The intersections of the vertical dashed line with the curves (shown by crosses) indicates the value of the basis function at  $X^{(k)}$  that multiplies the nodal fluid velocity in Eq. B.2 to compute the fluid velocity at  $X^{(k)}$ . . . . . 137

Figure 5.3 Contour plot of scaled analytical mean interphase momentum transfer term  $\langle F_x^{fp} \rangle / \langle F_x^{fp} \rangle_{\max}$  for the fluid velocity field given by Eq. 5.17 and mean particle velocity  $\langle V_1 \rangle = 1.0$ . . . . . 140

Figure 5.4 Convergence of forward interpolation error  $\epsilon_U$  with grid spacing  $h$ .  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The values in the legend are the slope of linear least-squares fit to the data. These are close to the order of the schemes. . . . . 146

Figure 5.5 Convergence of statistical error  $\text{rms}(\widehat{\Sigma}_F)$  with (a) number of multiple independent simulations  $\mathcal{M}$ , and (b) number of particles per cell  $N_{pc}$ , at  $(x = 0.5, y = 0.5)$  for Test 1 with zero particle velocity variance ( $V_1^* \stackrel{D}{=} \mathcal{N}[1.0, 0.0]$ ).  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The values in the legend are the slope of linear least-squares fit to the data. . . . . 148

Figure 5.6 Contour plot of total deterministic error for Test 1 on a  $21 \times 21 \times 4$  grid with  $N_{pc} = 400$  and  $N_{pc}\mathcal{M} = 60,000$ . The fourth-order LPI-4 scheme exhibits least error. All schemes show considerable spatial variation with an order of magnitude difference in the total deterministic error across the domain. . . . . 149

Figure 5.7 Estimation of bias coefficient  $b_F$  from plot of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, \mathcal{M}} \rangle_{\mathcal{M}}$  as a function of  $N_{pc}^{-1}$  for  $N_{pc}\mathcal{M} = 60,000$  at  $(x = 0.5, y = 0.5)$ .  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit which is also equal to bias coefficient is indicated in the legend. . . . . 151

Figure 5.8 Contour plot of the bias coefficient  $b_F$  for Test 1 on a  $21 \times 21 \times 4$  grid ( $N_{pc}\mathcal{M} = 60,000$ ). TSE incurs the least bias error with a bias coefficient that is two orders of magnitude lower than the other schemes. 152

Figure 5.9 Convergence of spatial discretization error  $S_F = \left| \{\mathbf{F}_m^{fp}\}_e - \langle \mathbf{F}_m^{fp} \rangle \right|$  with grid spacing  $h$  at different spatial locations to illustrate the strong spatial nonuniformity in convergence characteristics of the schemes: (a) convergence of  $S_F$  with  $h$  at  $(x = 0.5, y = 0.5)$  for Test 1, (b) convergence of  $S_F$  with  $h$  at  $(x = 0.6, y = 0.2)$  for Test 1.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend. . . . . 153

- Figure 5.10 Convergence of global discretization error  $\hat{S}_F$  with grid spacing  $h$  for Test 1.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend. Note the dramatic difference in convergence rates that deviate considerably from the order of the schemes, in contrast to Fig. 5.4. . . . . . 154
- Figure 5.11 Convergence of statistical error  $\text{rms}(\hat{\Sigma}_F)$  with (a) number of multiple independent simulations  $\mathcal{M}$ , and (b) number of particles per cell  $N_{pc}$ , at  $(x = 0.5, y = 0.5)$  for Test 2 with nonzero particle velocity variance ( $V_1^* \stackrel{D}{=} \mathcal{N}[1.0, 0.3]$ ).  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend. . . . . . 156
- Figure 5.12 Convergence of global discretization error  $\hat{S}_F$  with grid spacing  $h$  for Test 2.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least-squares fit is indicated in the legend. Comparison with Fig. 5.10 reveals that the effect of nonzero particle velocity variance on the magnitude and convergence rate of the global discretization error is minimal. 157
- Figure 5.13 Test 3 with nonuniform particle position distribution: (a) Contour plot of the scaled number density  $n(x, y)/n_o$ . (b) Contour plot of the scaled analytical mean interphase momentum transfer term  $\langle F_x^{fp} \rangle / \langle F_x^{fp} \rangle_{\max}$ . . . . . 158
- Figure 5.14 Ensemble-averaged mean interphase momentum transfer term for Test 3 with confidence intervals, and its error with respect to analytical value. In both panels the symbols indicate  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, \mathcal{M}} \rangle_{\mathcal{M}}$  (scale on left vertical axis) as a function of  $x$  at  $y = z = 0.5$ . Error bars denote 95% confidence intervals above and below the mean value, but are shown only above for clarity. The error bars for the two schemes in each panel are distinguished by the length of the cross-bars, with (a) LPI-4 (long), LPI-2 (short), and (b) PCA (long), TSE (short). Lines in both panels indicate the error  $\left| \langle \mathbf{F}_m^{fp} \rangle - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, \mathcal{M}} \rangle_{\mathcal{M}} \right|$  whose scale is given on the right vertical axis. . . . . . 159

Figure 5.15 Ensemble-averaged summed mean square error  $\hat{\epsilon}_F$  as a function of  $f_o$ , the frequency of the transcendental velocity field. . . . . 161

Figure 5.16 Contour plot of absolute total error  $\epsilon_F = \left| \{F_{x,m}^{fp}\}_{N_{pc},M} - \langle F_{x,m}^{fp} \rangle \right|$  for Test 2 with representative LE numerical parameters—a single realization  $\mathcal{M} = 1$  with  $N_{pc} = 5$  particles per cell on a  $61 \times 61 \times 4$  grid—to demonstrate the unacceptably large errors incurred by schemes such as LPI-4 and LPI-2. . . . . 165

Figure 6.1 Snapshot of a one-way coupled lid-driven cavity flow simulation at non-dimensional time  $tU/L$  equal to 10. Details are provided in Sec. 6.5. 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. . . . . 169

Figure 6.2 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  $h$ . . . . . 183

Figure 6.3 Convergence of Ensemble-averaged summed mean square error  $\hat{\epsilon}_F$  with number of particles per cell  $N_{pc}$  for (a) zero variance, and (b) finite variance in particle velocity distribution. The estimates have been obtained on a  $40 \times 40 \times 3$  grid using LPI-2 interpolation scheme. Number of independent realizations  $\mathcal{M}$  is equal to 100. . . . . 185

Figure 6.4 Contour plot of the root mean square statistical error  $\Sigma_F$  for the two cases of (a) zero, and (b) non-zero variance in particle velocity distribution. The number of particles per cell  $N_{pc}$  and number of independent realizations  $\mathcal{M}$  are equal to 5 and 100, respectively. The left and right panels in each subfigure correspond to the conventional and the improved estimator, respectively. . . . . 187

- Figure 6.5 Comparison of the convergence characteristics of the root mean square relative error  $\epsilon_F$  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. . . . . 189
- Figure 6.6 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. . . . . 190
- Figure 6.7 Contour plots of scaled analytical mean (a) number density  $\langle n_p \rangle / \langle n_p \rangle_{\max}$  and (b) interphase momentum transfer term  $\langle F_y^{\text{fp}} \rangle / \langle F_y^{\text{fp}} \rangle_{\max}$  for the test problem. . . . . 193
- Figure 6.8 Contour plots of the scaled mean interphase momentum transfer term  $\{F_y^{\text{fp}}\} / \langle F_y^{\text{fp}} \rangle_{\max}$  obtained from TLE1, TLE2, and ILE simulations of the test problem on  $30 \times 30 \times 3$  ((a), (c), and (e)) and  $100 \times 100 \times 3$  ((b), (d), and (f)) grids using LPI-2 with improved estimator. . . . . 196
- Figure 6.9 Contour plots of relative error  $\left( \left| \frac{\langle F_y^{\text{fp}} \rangle - \{F_y^{\text{fp}}\}}{\langle F_y^{\text{fp}} \rangle} \right| \right)$  obtained from TLE1, TLE2, and ILE simulations on  $30 \times 30 \times 3$  ((a), (c), and (e)) and  $100 \times 100 \times 3$  ((b), (d), and (f)) grids using LPI-2 with improved estimator. . . . . 198
- Figure 6.10 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.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA. . . . . 204
- Figure 6.11 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.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. . . . . 205

Figure 7.1	Schematic of two particles $i$ and $j$ having diameters $D_i$ and $D_j$ in contact. Particles have linear and angular velocities equal to $\mathbf{V}_i, \mathbf{V}_j$ and $\omega_i, \omega_j$ , respectively. Overlap $\delta_{ij} = 0.5(D_i + D_j) - D$ . $\eta_{ij}$ is the vector along the line of contact pointing from particle $i$ to particle $j$ . . . . .	210
Figure 7.2	Schematic of the spring–dashpot system used to model particle contact forces in soft–sphere approach. . . . .	211
Figure 7.3	2-D Schematic for “cell–linked list” neighbor search algorithm. Hollow and filled circles represent particles of different radii. . . . .	216
Figure 7.4	Comparison of the evolution of $h_{\max,k}/h_0$ (ratio of maximum height attained by a freely falling particle under gravity after $k$ collisions with wall to the initial height) with non–dimensional time, $t^* = t / \left( e_n \sqrt{\frac{2h_0}{g}} \right)$ , obtained from DEM simulation (denoted DEM) and analytical expression (denoted A) given by Eq. 7.33, for different values of coefficient of restitution $e_n$ . . . . .	218
Figure 7.5	Evolution of the relative error $\epsilon_h$ (Eq. 7.34) in estimation of $h_{\max,k}/h_0$ by the DEM simulations for different values of coefficient of restitution. . . . .	220
Figure 7.6	Schematic of the second verification problem. A spherical ball with finite translational velocity and zero angular velocity is placed on a rough surface. Forces acting on the ball is shown by the free body diagram on the right. . . . .	221
Figure 7.7	Comparison of $t' = \frac{\mu g t_s}{v_0}$ (left axis), and $\{v'_x, \omega'\}_{t=t_s}$ (right axis) obtained from DEM simulation with the expected values for different values of coefficient of friction. . . . .	222
Figure A.1	Particle position configurations obtained for the representative case ( $L/D = 8$ , $D = 1$ , $\varepsilon_s = 0.4$ ) after (a) lattice distribution (step 1), (b) shrinkage and mapping (step 2), and (c) elastic collisions (step 3). . . . .	238

- Figure A.2 Pair correlation function  $g(r)$  for different values of  $h_m$  (minimum grid cells between particle surfaces) obtained by averaging over 100 multiple realizations. In (a)  $\varepsilon_s = 0.01$ ,  $D_m = 10$ , and  $L/D = 15$ , and in (b)  $\varepsilon_s = 0.1$ ,  $D_m = 20$ , and  $L/D = 7.5$ . A safety factor of 1% (i.e.,  $X = 1$  in Eq. A.2) has been used. . . . . 239
- Figure B.1 1-D grid showing grid nodes and cells. . . . . 240

## ABSTRACT

The primary objective of this study is to improve the predictive capabilities of two-phase flow simulations that solve for average equations, such as Lagrangian–Eulerian (LE) and Eulerian–Eulerian simulations. The predictive capabilities of LE and EE simulations depend both on the numerical accuracy and on the accuracy of models for the fluid–particle and particle–particle interaction terms. In the first part of this study, a high fidelity ‘true’ DNS approach based on immersed boundary method (IBM) is developed to propose accurate models for fluid–particle terms, such as interphase momentum transfer, and also interphase heat and mass transfer, by solving for steady flow and scalar transport past homogeneous assemblies of fixed particles. IBM is shown to be a robust tool for simulating gas–solids flow and does not suffer from the limitations of lattice Boltzmann method (LBM): (1) compressibility errors with increasing Reynolds number; (2) calibration of hydrodynamic radius; (3) non-trivial to extend to non-isothermal systems. In the Stokes regime, average Nusselt number from scalar IBM simulations is in reasonable agreement with the frequency response measurements of Gunn and Desouza (1974) and free surface model of Pfeffer and Happel (1964), but differs by as much as 300% from the widely used heat and mass transfer correlation of Gunn (1978), which is attributed to the unjustified assumption of negligible axial diffusion in Stokes flow regime made by Gunn. At higher Reynolds numbers, scalar IBM results are far from Gunn’s correlations but in reasonable agreement with other experimental data. A correlation is proposed for heat and mass transfer as function of solid volume fraction and Reynolds for a particular value of Prandtl/Sherwood number equal to 0.7.

In the second part of this study, the numerical accuracy of LE simulations is investigated because LE simulations are very frequently used to verify EE simulations, and as a bench-

mark in the development of new simulation techniques for two-phase flows, such as the recent quadrature method of moments QMOM (Fox, 2008). Accurate calculation of the interphase transfer terms in LE simulations is crucial for quantitatively reliable predictions. Through a series of static test problems that admit an analytical form for the interphase momentum transfer term, it is shown that accurate estimation of the mean interphase momentum transfer term using certain interpolation schemes requires very high numerical resolution in terms of the number of particles and number of multiple independent realizations. Traditional LE (TLE) simulations, that use real particles or computational particles having constant statistical weight, fail to yield numerically-converged solutions due to high statistical error in regions with few particles. We propose an improved LE simulation (ILE) method that remedies the above limitation of TLE simulations and ensures numerically converged LE simulations.

## CHAPTER 1. INTRODUCTION

### 1.1 Background

A multiphase flow is a physical system in which several phases such as solid, liquid and gas can coexist, and these different phases affect each other hydro-dynamically. Multiphase flows are commonly observed ranging from the most natural occurrences such as rain drops in air, snowfall, volcanoes, and sandstorms to more practical applications in industries such as energy production, chemical processing, medicine and pharmaceuticals. Near the surface of the earth, particles are moved by interacting with air or water, which results in geological features affecting wide section of the population. Life-saving flu vaccine mists are delivered in the form of aerosols, or as a fine powder, to the human body. In internal combustion engines, finely-atomized fuel spray is injected into compressed air for efficient combustion and hence, less atmospheric pollution. A two-phase flow of a gas and a liquid is observed in oil-gas pipelines and wells, air-lift pumps, oil refineries, steam boilers. Needless to say, better understanding of the physical phenomena occurring in multiphase flows will help to make current applications more efficient and environmentally friendly. The scope of this study is limited to two-phase flows consisting of either solid particles or drops in a liquid or gas.

Unlike in laminar single-phase flows, one cannot meaningfully characterize a two-phase flow using only *one realization* (Drew, 1983; Pai, 2007). Also, for most real design purposes, engineers do not need the amount or detailed information that is generated from a single realization and, therefore, statistically averaged equations are mostly used to describe and solve two-phase flows. The most common statistical descriptions of two-phase flows can be classified into two broad categories: (i) Eulerian-Eulerian (EE) and (ii) Lagrangian-Eulerian (LE) representations. Only very recently, a Lagrangian-Lagrangian (LL) description has been

formulated by Pai (2007). In the EE representation, the two phases are assumed to be interpenetrating continua. A continuum description is adopted for both the carrier phase and the dispersed phase. Various averaging approaches have been applied to obtain the averaged conservation equations such as momentum and mass for each phase. The earliest averaging techniques to appear were time-averaging and space-averaging (and its variants based on the choice of averaging region) (Frankl, 1953; Teletov, 1958; Anderson and Jackson, 1967a; Drew, 1971) followed by the concept of ensemble averaging <sup>1</sup> (Drew, 1983; Kataoka and Serizawa, 1989; Drew and Passman, 1998).

In the LE statistical description (Williams, 1958; Subramaniam, 2000, 2001), the continuum description of the carrier phase is generally assumed to be identical to that in the EE representation. However, the dispersed phase is treated as composed of discrete entities in the system. These discrete particles <sup>2</sup> are statistically represented by a one-particle distribution function, which is often termed as droplet distribution function (ddf) in spray literature. The evolution of the ddf results in the famous spray equation (Williams, 1958). The state space in the spray equation is generally composed of position, velocity and radius co-ordinates, but, depending upon the physical problem, can have other variables, such as temperature, concentration, etc. Numerically solving the spray equation by traditional approaches such as finite difference or finite volume will require discretization for each state space, which even with today's computing power is impractical. Even though LE statistical description treats the solid-phase as discrete particles, by taking the moments of the evolution equation for one-particle distribution function, it is possible to derive continuum equations for the solid-phase as well. This is the approach taken in the kinetic theory of granular gases (KTGF) (Savage and Jeffrey, 1981; Garzo et al., 2007a), where the continuum equations for the solid-phase are derived from the one-particle distribution function and constitutive relations for the transport coefficients (such as the coefficients of viscosity, thermal diffusion coefficients, etc.) are obtained in terms of hydrodynamic variables (such as particle number density, inelasticity, granular temperature).

<sup>1</sup>Ensemble averaging is defined as the process of averaging a quantity over several independent realizations

<sup>2</sup>By particle we mean any dispersed-phase element, including solid particles, droplets and bubbles.

Since EE, LE, and LL statistical descriptions use different approaches to represent carrier and dispersed phases, the information content in each description can also be different. A detailed study investigating the consistency relationships between the three statistical descriptions is done by Pai (2007). The consistency relationship between the EE and LE statistical descriptions is investigated by comparing the fundamental quantities in each description: volume fraction and Eulerian velocity–radius joint pdf in EE description; number density and Lagrangian velocity–radius joint conditional pdf in LE description. It is concluded that the statistical information contained in the two descriptions is different and the aforementioned fundamental quantities are equal under very restrictive conditions, such as, statistically homogeneous number density and radius pdf. Also, the ddf is not able to capture the internal circulation effects in drops or bubbles. This poses a restriction on the class of physical problems that can be modeled by the ddf approach.

From the numerical viewpoint, and statistical descriptions notwithstanding, simulations that solve governing equations in both phases as continuous fields (i.e., continuum representation for both phases) will be referred to as EE simulations and those that consider carrier phase as continuum and dispersed phase as made up of discrete entities will be referred to as LE simulations. For example, as shown by the schematic in Fig. 1.1, EE simulations are possible from the averaged equations resulting from volume (Anderson and Jackson, 1967b) or ensemble averaging (Drew, 1971) approaches, and also from the one–particle distribution function (KTGF). There are plenty of open source and commercial codes available that are capable of doing both LE and EE simulations. For example, CFDlib (Kashiwa and Rauenzahn, 1994; Kashiwa and Gaffney, 2003) and MFIX (Syamlal et al., 1993; Syamlal, 1998) are both capable of performing EE simulations for chemically reacting multiphase flows. Similarly, examples of LE simulation codes are KIVA series of codes (A. A. Amsden, and P. J. O’Rourke, and T. D. Butler, 1989; Amsden, 1993) used widely in the automotive industry, MFIX–CDM code, and the commercially available Fluent code (discrete particle model). Very recently, yet another two–phase simulation method, termed as quadrature method of moments (QMOM), has been proposed by Fox (2008), which is briefly described below.

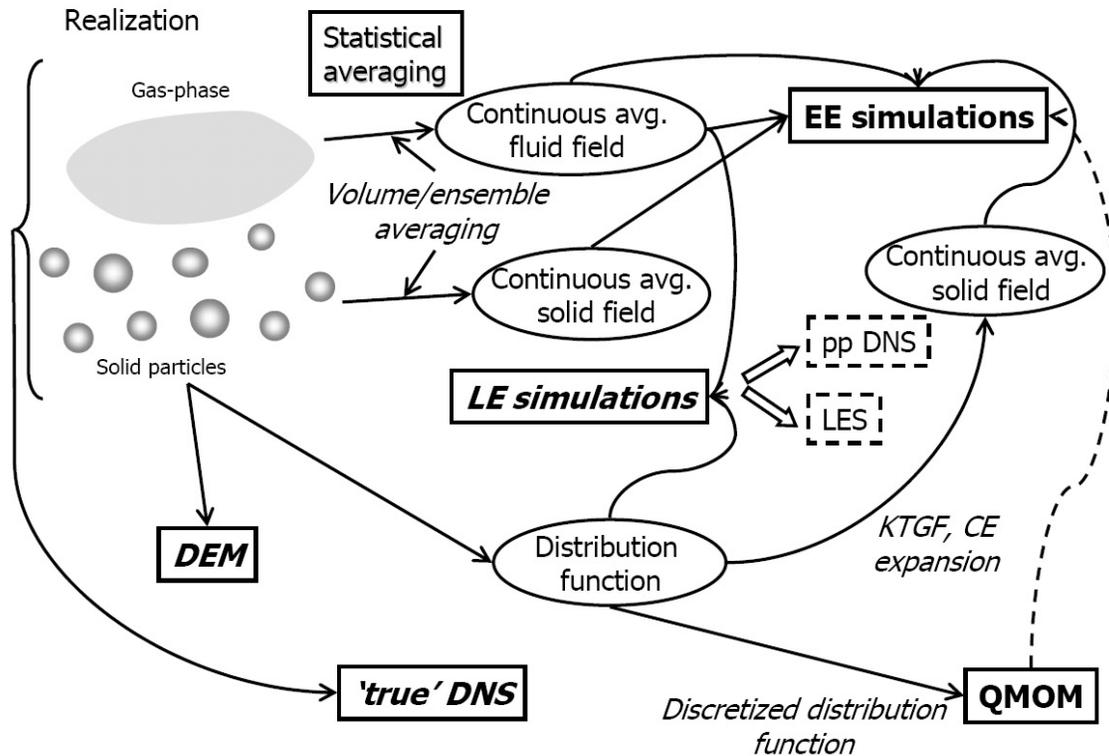


Figure 1.1 A schematic showing a realization of two-phase flow and various simulation types that can be used to solve for a realization ('true' DNS, discrete element method DEM) or the resulting averaged equations (LE, EE, and QMOM). The simulation types in ***bold italics*** are the ones considered in this work.

In QMOM, the carrier phase, like in LE and EE simulations, is considered as continuum. The solid-phase, like in LE statistical description, is represented by means of the one-particle distribution function. However, unlike in KTGF (moments of the one-particle distribution taken to arrive at continuous governing equations for solid-phase) and traditional LE simulations (one-particle distribution implied by ensemble of discrete particles), the discrete form of the one-particle distribution evolution equation (also called as kinetic equation) is numerically solved (see Fig. 1.1). This approach to solve for the kinetic equation directly is in principle a correct approach than EE simulations for non-equilibrium (i.e., non Maxwellian particle ve-

locity distribution) two-phase flows having high Knudsen number ( $Kn$ ) (defined as the ratio of mean free path to particle diameter). Since QMOM has been developed for two-phase flows, it inherently takes into account the additional physics (such as preferential concentration (Squires and Eaton, 1991) and particle trajectory crossing (Desjardins et al., 2008)) occurring due to fluid-particle interactions, such as the one due to non-conservative drag force acting on particles. In KTGF, however, only conservative body forces, such as the gravity force, are assumed to act on the particles. Although the resulting fields are continuous in both fluid-solid phases in QMOM, due to a very different approach taken to solve for the solid-phase, it is not termed as either LE or EE simulation in this introduction, and will be referred to as QMOM in the ensuing discussion.

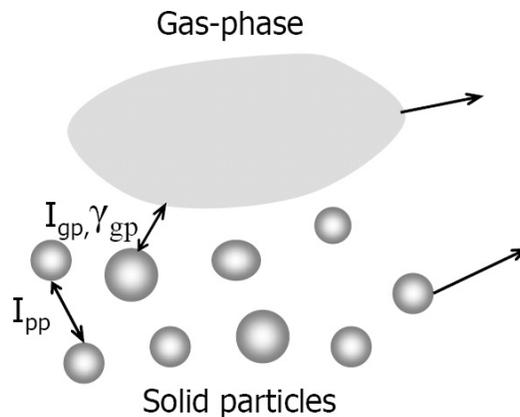


Figure 1.2 Schematic showing a cloud of gas-phase along with dispersed phase solid particles (solid-circles). In simulation solving for averaged equations (such as LE and EE), modeling of gas-solid interactions, such as interphase momentum transfer  $I_{gp}$ , interphase heat and mass transfer  $\gamma_{gp}$ , etc., and particle-particle interaction force  $I_{pp}$  are required.

So far we have discussed the simulations based on statistical representations (or averaged equations) of two-phase flows. However, one can also perform ‘true’ direct numerical simulation (‘true’ DNS) of a realization of two-phase flow system shown in Fig. 1.1. In ‘true’ DNS (see schematic in Fig. 1.1), the flow around each particle is solved with exact boundary conditions.

From the plethora of ‘true’ DNS methods available in literature, the most popular ones are the fictitious domain methods (Glowinski et al., 2001), immersed boundary methods (Peskin, 1981; Goldstein et al., 1993; Yusof, 1996), and Lattice–Boltzmann methods (Ladd, 1994b; Carte et al., 2004; van der Hoef et al., 2005). Although attractive, the downside with ‘true’ DNS techniques is the large computational cost which scales non–linearly with the Reynolds number. However, with the ever increasing computing power, accurate ‘true’ DNS has been, and will remain, a useful tool for comparing model accuracy for a wide variety of physical problems.

Similar to the ‘true’ DNS for gas–solids system, the discrete element method (DEM) is a widely used tool for numerical studies of granular flows. Some examples of the existing DEM based codes are LAMMPS code (Plimpton, 1995), MFIX–CDM code (Boylakunta, 2003), and the commercially available Arena flow code.

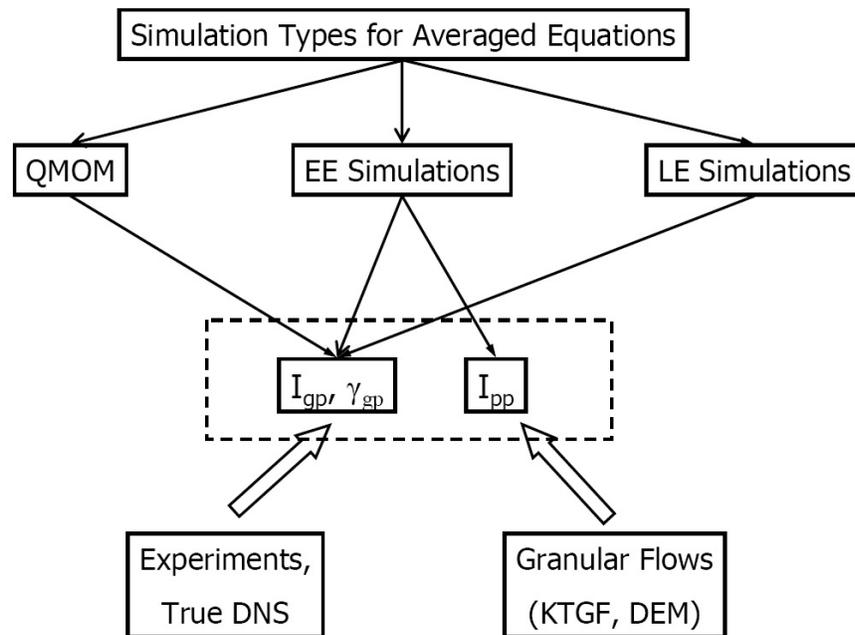


Figure 1.3 Schematic showing the modeling requirements (shown inside the dashed box) for two–phase flow simulations of averaged equations (LE, EE, and QMOM), and also the sources of models for fluid–particle and particle–particle interaction terms.

There are modeling and numerical implementation differences between LE, EE, and QMOM simulations. From the modeling viewpoint, the key differences between the simulation types lie in the modeling assumptions made in each approach. The schematic in Fig. 1.2 shows a realization of a cloud of gas-phase along with the dispersed phase particles (shown by solid-circles). If average equations are used to solve this fluid-particle system, then the modeling will be required for fluid-particle interactions, such as the interphase momentum transfer term  $I_{gp}$ , interphase heat and mass transfer term  $\gamma_{gp}$ , etc. Similarly, depending on the simulation type (LE or EE), further modeling may be required to model the particle-particle interaction force  $I_{pp}$  as well. As shown by the terms enclosed by dashed box in Fig. 1.3, in all simulations, fluid-particle interaction terms ( $I_{gp}$  and  $\gamma_{gp}$ ) have to be modeled. However, in EE simulations, additional models are required to describe the particle-particle interactions as well. The models for particle-particle interaction term needed in EE simulations are obtained from KTGF for low to moderate solid volume fractions, and numerical studies of granular flows for moderate to very dense (up to the close-packing limit) solid volume fractions.

Any simulation of averaged equations, LE, EE or QMOM, is only as good as the models that are used to describe the various interaction terms. One of the objectives of this study is to improve the accuracy of the existing models for various fluid-particle interaction terms, such as interphase momentum transfer, interphase heat and mass transfer, etc. In order to achieve this particular objective, a high fidelity ‘true’ DNS approach, using the immersed boundary method, is developed to solve for flow and scalar transport past homogeneous assemblies of fixed particles. IBM is a robust DNS tool as the increase in computational cost is very weakly dependent on the total number of particles, and one can solve complex geometries on structured Cartesian grids.

In the comparison of simulation methods for averaged equations, it was seen that relatively lesser modeling is required in LE simulation approach than in EE simulation approach. This is because in LE simulations, the particle-particle collisions can be explicitly treated through direct particle-particle collision models, such as hard-sphere (Allen and Tildesley, 1989) and soft-sphere (Cundall and Strack, 1978) collision models. It is worth noting that in LE simu-

lations of dilute systems, often times stochastic collision models, such as the droplet collision algorithm of O'Rourke (O'Rourke and Amsden, 1987) or the no time counter (NTC) (Bird, 1994; Schmidt and Rutland, 2000) collision algorithm, are used. The use of stochastic collision models, rather than direct models, essentially implies modeling of the particle-particle interaction term in LE simulations as well. However, the restriction imposed by the use of stochastic collision models on the range of validity of LE simulations (in terms of Knudsen number) is not as severe as the restriction on low Knudsen number imposed by KTGF closures in EE simulations. Therefore, due to relatively lesser modeling requirements, LE simulations are frequently used to verify EE simulations, and also used as a benchmark in the development of new two-phase flow simulations, such as the recent QMOM proposed by Fox (2008).

Although less modeling is required in the LE approach, it has its own limitations due to the use of discrete particles to represent the dispersed phase. LE simulations suffer from high statistical error resulting from the use of finite number of particles to compute mean quantities from particle data, such as the mean interphase momentum transfer term. For high Stokes number flows, the particle position distributions can become spatially very non-uniform; which further exacerbates the errors resulting from statistical noise. The problem of high statistical error is most severe in simulations of two-way coupled particle-laden two-flows where a poorly estimated interphase momentum transfer term when fed back to the flow field can lead to numerically non-converged results from LE simulations; thus, resulting in specious or erroneous conclusions. Since the quantitative comparison of LE simulation with any other simulation is a meaningful one when the numerical errors in both simulations are of the same order of magnitude, the need for accurate LE simulations cannot be over emphasized. Also, modeling error can only truly be assessed if the numerical error is low. The second part of this study investigates the numerical accuracy and convergence properties of the current LE simulations and proposes an improved LE simulation methodology that yields numerically converged and accurate results.

Having described the various numerical simulation methodologies for both average (LE, EE, and QMOM) and instantaneous equations ('pp' DNS, 'true' DNS, and DEM), and also

the different level of modeling requirements for average equations, we now move to the research objectives of this work.

## 1.2 Research Objectives

The principal objective of this study is to improve the predictive capabilities of two-phase flow simulations that solve for averaged equations, such as LE, EE, and QMOM. Among many modeling and numerical implementation details that affect the predictive capabilities of such simulations, we restrict ourselves to investigating the accuracy of fluid-particle interaction terms, such as interphase momentum transfer, and interphase heat and mass transfer terms. The objective is achieved by employing a two-pronged strategy. First the accuracy of existing correlations for fluid-particle interaction terms (such as drag law for interphase momentum transfer term and Nusselt number law for interphase heat transfer) are examined and new correlations proposed using ‘true’ DNS. Then the numerical accuracy of estimating fluid-particle interaction terms in LE simulations from finite number of particles is investigated and improved numerical schemes are developed. The research objectives are discussed below in more detail.

### 1.2.1 Modeling interphase momentum transfer term using ‘true’ DNS

The mean interphase momentum transfer term  $I_{gp}$  appears as a model term in all simulation types (Fig. 1.3) for two-phase flows. Therefore, accurate representation of the momentum transfer between the particles and fluid is absolutely necessary for predictive LE, EE, and QMOM simulations of two-phase flow. The dependence of this term on flow quantities such as the Reynolds number based on mean slip velocity, solid volume fraction, and particle size distribution must be modeled, and is simply referred to as a drag law.

The earliest attempts to provide drag law for fluid-particle interaction term were made by performing carefully controlled experiments. For example, Ergun (1952) studied the flow of gas through packed columns and provided the famous Ergun pressure drop correlation. With the advent of computers, it has now become possible to perform numerical simulations

of such complex systems, beginning from first-principles. True DNS of flow past particles is a first-principles approach to developing accurate models for interphase momentum transfer in gas-solids flow. The Lattice Boltzmann method (LBM) is an example of ‘true’ DNS. In LBM Ladd (1994a,b), instead of solving the Navier–Stokes equation, discrete one-particle velocity distribution function whose evolution is described by the lattice Boltzmann equation is solved. LBM has been successfully used to propose drag force correlations in static homogeneous assemblies of monodisperse (Hill et al., 2001a,b) and bidisperse (van der Hoef et al., 2005; Beetstra et al., 2007; Yin and Sundaresan, 2009) particles.

LBM is a highly efficient and robust solution methodology for gas-solids flow. LBM operations are local in physical space, it avoids solving the elliptic pressure Poisson equation that is needed in incompressible continuum flow solvers. This paves way for efficient parallelization of LBM, which has opened the door to solving realistic flow problems (Chen and Doolen, 1998). However, LBM simulations are limited by several limitations: (1) fine-tuning of the fluid coefficient of viscosity with increasing Reynolds number in order to keep the compressibility errors low; (2) calibration of hydrodynamic radius as a function of coefficient of viscosity and input radius, which results from stair-step of a spherical particle in LBM; (3) non-trivial to extend to non-isothermal systems.

In view of the above difficulties associated with LBM, we develop an alternative ‘true’ DNS methodology, termed as immersed boundary method (IBM). The basic notion of the immersed boundary method is to apply a set of forces on the computational grid to mimic the presence of an interface. The advantage of IBM is the favorable scaling of computational cost with the number of particles. IBM can be used to solve for flow around complex geometries on structured Cartesian grids, and there is no need for costly remeshing in the case of moving particles. There are many flavors of IBM existing in the literature that differ in the treatment of immersed boundary forcing. In this study, the discrete time immersed boundary method of Yusof (1996) has been extended to solve for flow past homogeneous assemblies of particles. An important improvement to the original IB forcing scheme used in Yusof (1996) has been proposed that ensures the fluid pressure and velocity fields are not contaminated by IB forcing.

The quantities computed from ‘true’ DNS using the IBM approach are related to the interphase momentum transfer term arising in theoretical approaches to gas–solids flow. This correspondence is described at different levels, starting from the one-particle distribution function and leading naturally to the averaged equation in that approach. An important connection of IBM quantities with two–fluid theory is also established.

In simulations of homogeneous assemblies, important numerical parameters are identified, such as grid resolution in terms of number of grid cells used to resolve a sphere diameter ( $D_m$ ), computational box length to particle diameter ratio ( $L/D$ ), number of particles  $N_s$ , and minimum number of multiple independent realizations  $\mathcal{M}$  required to ensure low statistical error in the estimate of drag force in random arrays (for ordered arrays, due to deterministic particle configuration, one realization is enough). The numerical convergence test results show that the IBM simulations yield grid-independent results, and these results are also independent of the choice of time step used to advance the solution in pseudo time, provided the stability criterion is met. A near second–order grid convergence is observed from IBM simulations. To place the numerical convergence study in context, we note that to our knowledge this is the most comprehensive study of numerical error and convergence for DNS of gas–solids flow.

The hydrodynamic IBM solver is extensively validated by comparing drag force from IBM simulations for three different cases: 1) comparison of drag force for flow past single particle in an unbounded medium with the single sphere drag correlation of Schiller and Naumann (1933), 2) comparison of drag force in Stokes flow regime for flow past SC and FCC arrangements (ranging from dilute volume fraction to close packed limit) with the boundary–integral method of Zick and Homsy (1982), and 3) comparison of drag force for moderate to high Reynolds ( $Re \leq 300$ ) in SC and FCC arrangements with the LBM simulations of Hill et al. (2001a). It is found that, wherever data is available, IBM simulations are in excellent agreement with the published data.

Extension of IBM to solve for steady flow past random arrays for  $0.01 \leq \varepsilon_s \leq 0.4$  and  $0 < Re \leq 300$  reveals an excellent match with LBM simulations of Hill et al. (2001b) and Beetstra et al. (2007) for low Reynolds number for both dilute and moderately dense random arrays.

However, the IBM simulations show a significant departure from these correlations at higher  $Re$ , and for dilute cases. The drag law proposed by Hill et al. (2001b) is stated to be more reliable for all Reynolds numbers only at higher volume fraction; therefore, the mismatch with their drag law for low to moderate volume fractions ( $\varepsilon_s < 0.1$ ) at high Reynolds numbers ( $Re > 100$ ) is justified. The mismatch with the drag law of Beetstra et al. (2007) is attributed to the coarse numerical resolutions used in their LBM simulations.

In order to generate random particle configurations, a three-step “random configurations initialization” algorithm has been developed. The three-step algorithm can be used to generate homogeneous random particle configurations up to very dense solid volume fractions ( $\approx 0.52$ ). The three-step algorithm has the ability to generate random arrays having the same volume fraction and number density, but differing in hard-core distance  $h_c$  (defined as the minimum distance between the centers of any two particles). It is shown that much higher values for hard-core distance are accessible through the three-step algorithm than those possible from stochastic methods, such as the Matérn point-process (Stoyan and Stoyan, 1995).

IBM is a robust simulation method for ‘true’ DNS of gas–solids flow and does not suffer from the above identified difficulties in LBM. With the successful extension of discrete time immersed boundary method of Yusof (1996) to solve for flow past homogeneous assemblies for arbitrary values of solid volume fraction  $\varepsilon_s$  and Reynolds number  $Re$ , it has now become possible to extend the versatile IBM approach to include additional physics. For example, current efforts are ongoing to include effects of gas-phase turbulence (Xu, 2008), polydispersity in the size distribution of solid particles, and transport of chemical species and heat due to fluid flow to the IBM solver. In this study, the hydrodynamic IBM solver is extended to study passive scalar transport and heat transfer past random and ordered arrays of spheres, which is our next research objective: development of accurate closures for the mean interphase heat and mass transfer term  $\gamma_{gp}$  through ‘true’ DNS.

### 1.2.2 Modeling mean interphase heat and mass transfer term using ‘true’ DNS

The mean interphase heat and mass transfer term  $\gamma_{gp}$ , just like the mean interphase momentum transfer term  $I_{gp}$ , also appears as a model term in all simulation types (Fig. 1.3) for two-phase flows. Therefore, accurate dependence of this term on flow quantities such as the Reynolds number based on mean slip velocity, solid volume fraction, particle size distribution, and Prandtl (or Sherwood number for mass transfer) number must be modeled for predictive LE and EE simulations of gas-solids flow in industrial applications. Due to heat and mass transfer analogy, the discussion below is motivated as a heat transfer problem, but the conclusions of the study apply equally to mass transfer as well.

In two-phase flow simulations, the interphase heat transfer term is typically closed by using a correlation for the Nusselt Nu number. There is a plethora of experimental literature spanning over the last seven decades (see Wakao and Kaguei (1982) for a comprehensive review) on heat and mass transfer in packed gas-solid and liquid-solid fixed-bed reactors. Various experimental techniques, such as axial heat conduction in beds (Kunii and Smith, 1961), step response (Handley and Heggs, 1968), frequency response (Gunn and Desouza, 1974; Littman et al., 1968), and shot response (Shen et al., 1981; Wakao et al., 1977), have been used to study heat and mass transfer in fixed-beds. Although, several correlations have been proposed for heat and mass transfer in gas-solids flow from the experimental studies, the average Stokes Nusselt number  $Nu_0$  (defined as Nusselt number in the Stokes flow regime) has been a issue of much controversy.

These experimental studies have reported values for the average Stokes Nusselt number that differ by orders of magnitude. Some studies suggested average Stokes Nusselt number in packed beds to be less than 2 (which is equal to the Nusselt number for an isolated sphere at zero Re) and tending to zero (Kunii and Smith, 1961; Littman et al., 1968; Cybulski et al., 1975), while others reported values greater than 2 and as high as 13 (Pfeffer and Happel, 1964).

In view of so much disagreement over the Stokes Nusselt number value in experimental studies, we wish to numerically study heat and mass transfer. True DNS is a convenient and robust tool for numerically studying heat and mass transfer in fixed and fluidized beds. In this

work, heat transfer is studied in fixed beds by simulating steady Stokes flow past homogeneous arrays of monodisperse spherical particles. Direct forcing immersed boundary method (IBM) originally developed by Yusof (1996), and extended in this work to simulate steady flow past homogeneous assemblies, is further extended to solve for passive scalar transport past homogeneous particle assemblies. An isothermal boundary condition is assumed for the particle surface temperatures. Heat conduction within the particles is not considered. Furthermore, the feedback of scalar transport on the hydrodynamic fields due to density variation (free convection effects) is not considered in this study. These assumptions limit the gas–solid systems that our simulations can be applied to. The regime of validity of our simulations for both heat and mass transfer given the above assumptions is discussed in Sec. 3.2.1.

A novel method of simulating passive scalar transport in homogeneous assemblies is developed. Due to the periodic boundary conditions used for homogeneous flows and isothermal boundary condition for the particle surface, the fluid scalar field will eventually equilibrate to solid surface value, resulting in steady–state with zero scalar flux across the particle–fluid interface. In order to obtain steady–states with finite scalar flux across the particle–fluid interface, a source/sink term is added to the scalar transport equation. The source/sink term in the scalar transport is shown to be an analog of the mean pressure gradient term in momentum transport. As for the hydrodynamic case, an important connection of IBM quantities with two–fluid theory is established for heat and mass transfer as well.

Numerical convergence tests are performed to demonstrate that the scalar IBM simulations yield grid-independent results, and that these results are also independent of the choice of time step used to advance the solution in pseudo time step. The scalar IBM solver is validated against the Nusselt number law for isolated particle (Ranz and Marshall, 1952) in an unbounded medium.

Scalar IBM simulations are first performed to obtain the average Stokes Nusselt number as a function of solid volume fraction for both ordered and random arrays. It is found that for random arrays, scalar IBM results are in reasonable agreement with the frequency response measurements of (Gunn and Desouza, 1974) and the free surface model of (Pfeffer and Happel,

1964), but differs by as much as 300% from the widely used heat and mass transfer correlation of (Gunn, 1978) for the highest solid volume fraction of 0.5 that is considered in this study. The large differences between scalar IBM simulations are primarily attributed to the neglect of axial diffusion in Gunn's stochastic model (Gunn, 1978). Through a budget study of the scalar transport equation, it is shown that axial diffusion is one-third of the total diffusion.

The scalar fields from scalar IBM simulations are used to examine local Nusselt number along the surface of the particle. It is found that in the Stokes flow regime, the peaks of local Nusselt number (implying locally highest heat transfer) occur in the regions of maximum flow channel width. As the Reynolds number increases, the peak of local Nusselt number shifts from the regions of maximum flow channel width in Stokes flow regime toward the regions of minimum flow channel width. This is attributed to the dominance of convective transport over the diffusive transport as the Reynolds number increases. Since by mass continuity, the fluid velocity is highest in narrow regions between particles, therefore, in the convective transport dominated regime, the maximum heat transfer, and, thus, the maximum local Nusselt number, shifts toward the regions of narrow flow channel widths.

The extension of scalar IBM simulations to higher Reynolds number reveals that scalar IBM results are far from Gunn's correlations but in reasonable agreement with other experimental data. A correlation is proposed for heat and mass transfer as function of solid volume fraction and Reynolds for a particular value of Prandtl/Sherwood number equal to 0.7.

The new correlation for heat and mass transfer proposed from IBM simulations finishes the first principle objective of this study. Next, we turn our attention to the second principle objective of this study, which is improving the numerical accuracy and convergence properties of LE simulations.

### 1.2.3 Numerical Accuracy and Convergence Characteristics of LE simulations

LE simulations are frequently used to verify EE simulations, and also used as a benchmark in the development of new simulation methods for two-phase flows. However, comparison between the any two simulation types is meaningful only if the numerical errors in both

simulations are of the same order of magnitude. In the context of numerical accuracy and convergence characteristics of LE simulations, we ask the following questions:

1. What is the numerical accuracy of estimating interphase source terms, such as, momentum transfer, energy transfer, etc., from a finite number of discrete particles ?
2. What are the numerical convergence characteristics of LE simulations ?

Many studies have demonstrated the non-convergence of popular numerical implementations of the LE approach to spray modeling (Subramaniam and O'Rourke, 1998; Iyer and Abraham, 1997; Aneja and Abraham, 1998), and also to gas-solid fluidized beds (Sun et al., 2007). A systematic investigation of the numerical convergence characteristics of the KIVA implementation of the LE approach led to the conclusions that even global spray characteristics such as spray penetration length did not show any trend toward convergence to an asymptotic value as the numerical parameters such as grid size, time step, and computational particles were varied toward their limiting values (Subramaniam and O'Rourke, 2001). In LE simulations, the mean interphase transfer terms, such as momentum, energy, mass, etc., are calculated on an Eulerian grid from a finite number of particles. Accurate calculation of interphase transfer terms is absolutely vital to the overall accuracy of two-way coupled particle-laden flow simulations. In this context, pertinent questions that this work will attempt to answer are:

1. What is the accuracy of various numerical schemes in calculating interphase transfer terms?
2. Is it possible to obtain numerically (grid) converged estimates for interphase transfer terms?

In order to address the first question, a series of static test problems that admit an analytical solution for the mean interphase momentum transfer term are used to assess the accuracy of popular estimation schemes. An error model similar to the one proposed in Xu and Pope (1999) is used to decompose the error in estimating mean interphase momentum transfer term into various components. The error is characterized as a function of number of particles, grid

resolution, and number of multiple but independent simulations. It is observed that for the resolution (in terms of the number of particles) used in typical LE simulations, the error in estimating mean interphase momentum transfer term is as high as 80%.

In traditional LE (TLE) simulations, either real particles or statistically weighted computational particles are used to represent the dispersed phase. The statistical weight implies the expected number of real particles represented by each computational particle. Traditionally, the particles are weighted equally and their weights do not evolve in time. Additionally, the number of computational particles do not scale with the grid size. With the help of time-evolving test problem, this study demonstrates the inability of such TLE simulations to yield numerically converged estimates for the mean interphase momentum transfer term. An improved LE (ILE) approach wherein the statistical weights of computational particles is evolved in time in order to ensure their near uniform spatial distribution at all times is proposed in this study. Numerical estimates of the mean interphase momentum transfer term are compared from 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 a slightly modified conventional estimator, referred to as the improved estimator. The improved estimator results in more accurate and faster converging estimates than the conventional estimator. The ILE simulation method along with the improved estimator is shown to result in more accurate and, as well as, numerically-converged LE simulations.

It is worth mentioning that the implications of our findings and improvements to LE method are equally applicable to any particle-based LE simulations, such as DNS and LES of two-phase flows that represent the dispersed phase as point sources. Accurate LE simulation method, such as ILE, can be reliably used to test new sub-models for particle-particle interaction term in EE simulations, and serve as a benchmark tool in the development of new simulation techniques (such as QMOM).

### 1.2.4 Development of an open source DEM code for simulations of granular and gas-particle systems

For gas-particle flows, there are many codes available, both commercial (Fluent) and open source (CFDlib, MFIx), that can solve the averaged continuum equations (i.e. both phases represented as continua). Since the discrete particle model currently available in Fluent considers only the stochastic collisions (O'Rourke and Amsden, 1987), it is therefore limited to describing dilute systems. Similarly, for granular flows, there are commercial and open source codes available that one can use to simulate. However, MFIx provides a single source code having the capability to solve equations both for the continuum and the discrete descriptions that can be used to simulate gas-particle systems, as well as, granular flows.

In the MFIx code, a basic structure for DEM simulations has now existed for a couple of years. However, MFIx is not as widely used for DEM simulations as it is for continuum simulations even though it is an excellent opportunity to be able to run different descriptions from one platform. The reason behind MFIx-DEM modules lesser usage is its lack of reliability. Besides, the original MFIx-DEM modules were good only for dense and monodisperse systems.

The MFIx-CDM code over the last one year has been extensively debugged and two new features have been added: 1) ability to run dilute systems, and 2) ability to run any particle size/density distribution. In order to make it suitable to run dilute systems, a suite of interpolation routines has been added in order to calculate the drag force on each particle and also its reverse projection onto the Eulerian grid (details in chapter 5).

## 1.3 Thesis outline

In Chapter 2, details of the hydrodynamic IBM solver are presented. The quantities computed from 'true' DNS using the IBM approach with the interphase momentum transfer term arising in theoretical approaches to gas-solids flow. This correspondence is described at different levels, starting from the one-particle distribution function and leading naturally to the averaged equation in that approach. An important connection of IBM quantities with two-fluid theory is also established. Numerical convergence and validation of the hydrodynamic

IBM solver are discussed. Drag force from IBM simulations for flow past ordered and random arrays is compared with the existing the monodisperse drag correlation from LBM simulations.

In Chapter 3, the hydrodynamic IBM solver is extended to solve for passive scalar transport past ordered and random arrays. Scalar IBM governing equations are related to the averaged equations solved in the two-fluid theory. Numerical convergence and validation of the scalar IBM solver are discussed. Average Stokes Nusselt obtained from scalar IBM simulations for ordered and random arrays is compared with Gunn's correlation. Local Nusselt number and local viscous drag are examined along the particle surface in ordered arrays in order to explain the asymptote of average Stokes Nusselt in the limit of close-packing.

In Chapter 4, the scalar IBM simulations are extended to high Reynolds number up to  $Re = 300$  for both ordered and random arrays. Average Nusselt number obtained from scalar IBM simulations is compared with Gunn's correlation and a new correlation proposed for heat and mass transfer in gas-solids flow.

In Chapter 5, the accuracy of popularly used interpolation schemes in estimating the mean interphase momentum transfer term is examined. A series of static test problems that admit an analytical solution for mean interphase momentum transfer term are performed to systematically characterize the effects of varying the particle velocity variance, the distribution of particle positions, and fluid velocity field spectrum on estimation of the mean interphase momentum transfer term. Numerical error resulting from backward estimation is decomposed into statistical and deterministic (bias and discretization) components, and their convergence with number of particles and grid resolution is characterized.

In Chapter 6, an improved LE simulation (ILE) method for volumetrically dilute flow particle-laden flows is proposed. In addition, an improved estimator is also proposed for more accurate estimation of mean interphase momentum transfer term. Starting from the one-particle density function, consistency relationships are derived for statistical equivalence to hold between TLE and ILE simulations.

Chapter 7, implementation details of the MFIx-DEM code are provided. This document, when completed, will serve as a theory guide and user manual to the future users of MFIx-

CDM code.

## CHAPTER 2. DIRECT NUMERICAL SIMULATION OF GAS-SOLIDS FLOW BASED ON THE IMMERSED BOUNDARY METHOD

This chapter is a manuscript in review titled “Direct numerical simulation of gas-solids flow based on the immersed boundary method” due to appear as a chapter in the book titled “Computational gas–solid flows and reacting systems: theory, methods and practice” edited by S. Pannala, M. Syamlal and T. J. O’Brien. The authors for this chapter are R. Garg, S. Tenneti, J. M. Yusof, and S. Subramaniam.

### 2.1 Introduction

Accurate representation of the momentum transfer between particles and fluid is necessary for predictive simulation of gas-solids flow in industrial applications. Such device-level simulations (Syamlal et al., 1993) are typically based on averaged equations of mass and momentum conservation corresponding to the fluid and particle phase(s) in gas-solids flow, and these constitute the multi-fluid theory. The momentum conservation equation in this theory contains a term representing the average interphase momentum transfer between particles and fluid. The dependence of this term on flow quantities such as the Reynolds number based on mean slip velocity, solid volume fraction, and particle size distribution must be modeled in order to solve the set of averaged equations, and is simply referred to as a drag law. If higher levels of statistical representation are adopted – such as for the second moment of particle velocity, or for the particle distribution function – then the corresponding terms (such as the interphase transfer of kinetic energy in the second velocity moment equations) appearing in those equations also need to be modeled.

Direct numerical simulation of flow past particles is a first-principles approach to developing

accurate models for interphase momentum transfer in gas-solids flow at all levels of statistical closure. While there are different numerical approaches available to perform DNS of gas-solids flow - such as the lattice Boltzmann method (LBM) - here we describe a DNS approach that is based on the immersed boundary method (IBM).

### 2.1.1 Transport of the particle distribution function

The transport equation for the one-particle distribution function in gas-solids flow for monodisperse case is (Chapman and Cowling, 1953; Koch, 1990; Jenkins and Savage, 1983; Subramaniam, 2001; Garzo et al., 2007a)

$$\frac{\partial f}{\partial t} + \nabla_x \cdot (\mathbf{v}f) + \nabla_v \cdot (\langle \mathbf{A} | \mathbf{x}, \mathbf{v}, \mathbf{t} \rangle f) = \dot{f}_{\text{coll}}, \quad (2.1)$$

where  $\dot{f}_{\text{coll}}$  is the particle-particle collision term. The principal difference between this equation for solid particles and its counterpart in molecular gases is the appearance of the conditional expectation of the acceleration inside the velocity derivative corresponding to transport of the distribution function in velocity space. The conditional expectation of acceleration cannot be expressed purely in terms of the distribution function, and is hence denoted an unclosed term in the above equation. It can depend on higher-order distribution functions (e.g., the two-particle distribution function) in the hierarchy resulting from a description of the particle system in terms of the Liouville density. It also depends on statistics of the carrier flow. Since analytical models are difficult to propose for this term beyond dilute particle flow in the Stokes flow regime, it must be inferred from direct numerical simulation data. Drag laws for steady flow through homogeneous suspensions are obtained by integrating the conditional expectation of the acceleration over velocity space to obtain the average force  $\langle \mathbf{F}_d \rangle$  exerted on the particles by the fluid

$$\langle \mathbf{F}_d \rangle = \frac{m}{n} \int \langle \mathbf{A} | \mathbf{v} \rangle f \mathbf{d}\mathbf{v} \quad (2.2)$$

where  $m$  is the mass of a particle, and  $n$  is the particle number density.

### 2.1.2 Homogeneous suspension flow

In order to calculate  $\mathbf{F}_d$  from DNS, it is natural to simulate a statistically homogeneous suspension flow with freely moving particles, and to then compute volume-averaged estimates of  $\mathbf{F}_d$  from particle acceleration data. Imposing a mean pressure gradient to balance the weight of the particles leads to a steady mean momentum balance. In this setup the particle positions and velocities sample a trajectory in phase space that corresponds to the specified nonequilibrium steady state, and time averaging can be used to improve the estimate for  $\mathbf{F}_d$ . However, such freely moving suspensions are computationally prohibitive especially because in order to propose drag laws these simulations need to be performed over a range of solid volume fractions and mean flow Reynolds numbers (based on mean slip velocity). Furthermore, over a wide range of volume fraction and particle Stokes number, the particle configuration in individual realizations develops spatial structures due to flow instabilities. Wylie and Koch (2000) performed simulations of a suspension with particles moving along ballistic trajectories between elastic hard-sphere collisions, but this assumption that the fluid does not affect the particle motion is valid only in the limit of high Stokes number.

Koch and Hill (2001) discuss the relevant non-dimensional parameters that arise in the context of gas-solid suspensions. As noted in their work, direct numerical simulations are useful in developing drag laws for suspension flows where the effects of fluid inertia and the particle inertia cannot be neglected. In the simulations described in this work we neglect gravity, so the relevant nondimensional parameters are the Reynolds number (characterizing the importance of fluid inertia) and the particle Stokes number (characterizing the importance of particle inertia). While the Stokes flow regime (negligible fluid inertia) is amenable to analytical treatment, direct simulation is the only approach for gas-solid suspensions at finite Reynolds number.

### 2.1.3 Steady flow past homogeneous assemblies of fixed particles

A convenient simplification for high Stokes number suspensions is to replace the ensemble of particle positions and velocities sampled by the system in its non-equilibrium steady state,

by a set of particle configurations and velocities that would result from a granular gas simulation. Steady flow past fixed assemblies of particles in configurations (and with velocities) sampled from this set is simulated, and drag laws are obtained by averaging over this ensemble. The idea of extracting drag laws from steady flow past random and ordered arrays of particles through particle assemblies has been successfully exploited by several researchers using the LBM simulation methodology developed by Ladd (1994a,b) for particulate suspensions. For example, Koch and co-workers (Hill et al. (2001a) and Hill et al. (2001b)), referred to collectively as HKL, studied the steady flow past both ordered and random arrays. Kuiper's and co-workers (van der Hoef et al. (2005) and Beetstra et al. (2007)), collectively referred to as BVK, extended HKL's LBM simulations to higher Reynolds numbers.

In the simplest case of a monodisperse suspension, the drag law is extracted by computing steady nonturbulent flow at a specified mean slip Reynolds number past a set of random particle configurations (microstates) that correspond to a particular value of the solid volume fraction. The pair-correlation and higher-order statistics of the particle field are determined by the configurations resulting from the granular gas simulation. The particle velocity distribution can be initialized either from the granular gas simulation at finite granular temperature or it is often assumed that all particles move with the same velocity.

## 2.2 Governing equations

The schematic in Fig. 2.1 corresponds to the physical problem of flow past a single particle. Volumes occupied by the fluid and solid phases are denoted by  $\mathcal{V}_f$  and  $\mathcal{V}_s$ , respectively, such that the total domain volume  $\mathcal{V} = \mathcal{V}_f + \mathcal{V}_s$ . The bounding surfaces of the physical domain, solid-phase, and fluid-phase are denoted by  $\partial\mathcal{V}$ ,  $\partial\mathcal{V}_s$ , and  $\partial\mathcal{V}_f$ , respectively. For incompressible flows, the mass and momentum conservation equations for the fluid-phase are

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (2.3)$$

and

$$\rho_f \frac{\partial u_i}{\partial t} + \rho_f \frac{\partial u_i u_j}{\partial x_j} = -g_i + \mu_f \frac{\partial^2 u_i}{\partial x_j \partial x_j} = \frac{\partial \tau_{ji}}{\partial x_i}, \quad (2.4)$$

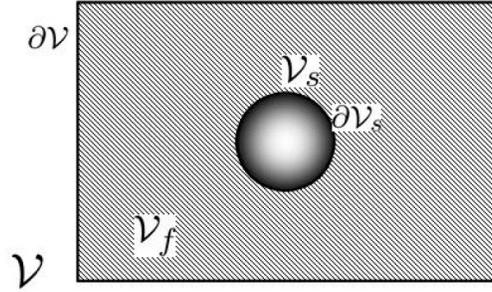


Figure 2.1 ]A simple schematic of the physical domain with only one particle. Hatched lines represent the volume  $\mathcal{V}_f$  occupied by the fluid-phase and solid fill represents the volume  $\mathcal{V}_s$  of the solid-phase such that the total volume of physical domain  $\mathcal{V} = \mathcal{V}_f + \mathcal{V}_s$ . The bounding surfaces of the physical domain, solid-phase, and fluid-phase are denoted by  $\partial\mathcal{V}$ ,  $\partial\mathcal{V}_s$ , and  $\partial\mathcal{V}_f$ , respectively.

respectively, where  $\mathbf{g}$  is the pressure gradient, and  $\rho_f$  and  $\mu_f$  are the thermodynamic density and dynamic viscosity of the fluid-phase, respectively. At the particle-fluid interface, in order to ensure zero slip and zero penetration (for impermeable surfaces) boundary conditions, the relative velocity should be zero. If the solid particles are held stationary, then the above boundary conditions translate to

$$\mathbf{u}_f = 0 \quad \text{on} \quad \partial\mathcal{V}_s, \quad (2.5)$$

The averaged equations corresponding to these mass and momentum conservation balances are useful in simulations of practical gas-solids flow applications. In the earlier section we described one statistical approach based on the one-particle distribution function. Here we first describe an alternative approach called the Eulerian two-fluid theory because it is more natural to derive the averaged equations corresponding to Eq. 2.4 using this approach. The conditional expectation of acceleration appearing in the one-particle distribution function approach is then related to the mean interphase momentum transfer term in the Eulerian two-fluid theory.

In the Eulerian two-fluid theory phasic averages are defined as follows. If  $Q(\mathbf{x}, t)$  is any field, then its phasic average  $\langle Q^{(f)} \rangle(\mathbf{x}, t)$  over the fluid volume  $\mathcal{V}_f$ , referred to as fluid-phase

mean, is defined as:

$$\langle Q^{(f)} \rangle(\mathbf{x}, t) = \frac{\langle I_f(\mathbf{x}, t) Q(\mathbf{x}, t) \rangle}{\langle I_f(\mathbf{x}, t) \rangle}, \quad (2.6)$$

where  $I_f(\mathbf{x}, t)$  is an indicator function which is one if the point  $\mathbf{x}$  lies on the fluid-phase and zero otherwise.

The solid-phase mean  $\langle Q^{(s)} \rangle(\mathbf{x}, t)$  is similarly defined. The (unconditional) mixture mean  $\langle Q \rangle$  is related to the phasic means by:

$$\langle Q \rangle = \varepsilon_f \langle Q^{(f)} \rangle + \varepsilon_s \langle Q^{(s)} \rangle, \quad (2.7)$$

where  $\varepsilon_f = \langle I_f \rangle$  and  $\varepsilon_s = \langle I_s \rangle$  are the volume fractions of the fluid and solid phases, respectively. If the flow is statistically homogeneous, there is no dependence on  $\mathbf{x}$ , and spatial derivatives are zero. Similarly if the flow is statistically stationary, there is no dependence on  $t$ , and temporal derivatives are zero.

The mean momentum conservation equation in the fluid phase (Drew, 1983; Pai and Subramaniam, 2008) is obtained by multiplying the momentum conservation equation 2.4 by  $I_f$ , resulting in

$$\frac{\partial \rho_f \varepsilon_f \langle u_i^{(f)} \rangle}{\partial t} + \frac{\partial}{\partial x_j} \rho_f \varepsilon_f \langle u_i^{(f)} \rangle \langle u_j^{(f)} \rangle = \frac{\partial}{\partial x_j} \langle I_f u_i''^{(f)} u_j''^{(f)} \rangle + \left\langle I_f \frac{\partial \tau_{ji}}{\partial x_j} \right\rangle, \quad (2.8)$$

where  $u_i''^{(f)} = u_i - \langle u_i^{(f)} \rangle$  is the fluctuating component of the fluid velocity field. For steady flow with an imposed mean pressure gradient in the fluid phase, it is convenient to decompose the pressure gradient term that appears in the divergence of the fluid phase stress tensor as  $\mathbf{g} = \langle \mathbf{g} \rangle + \mathbf{g}'$ , such that remaining part of the stress tensor  $\tau'_{ji}$  is defined by the expression:

$$\frac{\partial \tau_{ji}}{\partial x_j} = -\langle g_i \rangle - g'_i + \mu_f \frac{\partial^2 u_i}{\partial x_j \partial x_j} = -\langle g_i \rangle + \frac{\partial \tau'_{ji}}{\partial x_j}. \quad (2.9)$$

For a statistically homogeneous suspension at steady state (statistically stationary flow), the average velocity does not depend on  $\mathbf{x}$  or  $t$ , and the unsteady and convective terms on the left hand side of Eq. 2.4 do not contribute. Writing the remaining terms in an integral form shows that the mean pressure gradient term balances the sum of fluctuating pressure and viscous stress on the solid particles:

$$\varepsilon_f \langle g_i \rangle = - \left\langle \tau'_{ji} n_j^{(s)} \delta(\mathbf{x} - \mathbf{x}^{(I)}) \right\rangle \quad (2.10)$$

In the above equation  $n_j^{(s)}$  is the normal vector pointing outward from the particle surface into the fluid, and the stress tensor is evaluated on the fluid side of the interface. This term appears as the drag contribution  $F_{\text{gm}}(\mathbf{v}_{\text{sm}} - \mathbf{v}_{\text{g}})$  to the fluid-solids interaction force  $I_{\text{gm}}$  term, and it is the equivalent expression for Eq. 2.10 in the two-fluid equations derived from a volume-averaging approach (Syamlal et al., 1993). For statistically homogeneous flows, the relationships between the one-particle distribution function approach and the Eulerian two-fluid theory are established in the context of a comprehensive probability density function approach to multiphase flows (Pai and Subramaniam, 2008). Using the relationships in Pai and Subramaniam (2008), it is easy to show that the term on the right hand side of Eq. 2.10 is related to the average force exerted by the fluid on the particles as follows:

$$\langle F_{d,i} \rangle = m \langle A_i \rangle = \frac{1}{n} \left\{ -\varepsilon_s \langle g_i \rangle + \left\langle \tau'_{ji} n_j^{(s)} \delta(\mathbf{x} - \mathbf{x}^{(I)}) \right\rangle \right\} \quad (2.11)$$

### 2.3 The immersed boundary method

The basic notion of the immersed boundary method is to apply a set of forces on the computational grid to mimic the presence of an interface. This has several advantages over conventional boundary-fitted grids, especially for problems involving moving interfaces. First, there is no overhead for grid generation, which can consume considerable computational effort even for fixed geometries. Second, the convergence of the solvers is generally improved for Cartesian meshes. Third, these meshes require much less storage overhead than general unstructured or curvilinear meshes. The primary disadvantage of IBM is the reduced resolution near the interface, but this is remedied by adopting adaptive mesh techniques. There are two basic facets of the IBM, namely the choice of flow field (i.e. what velocity field do we wish to achieve) and calculation of the force itself (i.e. once we decide on the field we wish to achieve, how do we specify the force at each time-step). For clarity we will separate these two aspects, dealing with the force specification first.

The immersed boundary method was originally developed by Peskin (1981) as a way to incorporate the effect of flexible interfaces into fluid simulations. In that version, the local force is obtained from some constitutive relation commensurate with the nature of the interface (e.g

surface tension in the case of a bubble, Youngs modulus for an elastic membrane) and is, by necessity, iterative over a timestep since the location of the interface is not known *a priori*. This method has been applied to a variety of flows, such as bubbles, blood cells and swimming fish. The issue with this implementation is that it is not efficient for rigid bodies, since this requires driving the stiffness of the interface membrane (and effectively the stiffness of the equations to be solved) to infinity.

Goldstein et al. (1993) proposed what is essentially proportional-integral feedback on the force term to produce boundary conditions on a rigid body. The problem with this method is the lack of efficiency; due to the need to numerically integrate the force in (pseudo-continuous) time over a single time-step, the effective CFL limit was extremely small,  $\mathcal{O}(10^{-3})$ . Coincident with Goldsteins work, Yusof (1996) developed what is now termed the Discrete-Time Immersed Boundary Method (DTIBM). The essential aspect of this formulation is the recognition that examination of the discretized-in-time equations leads to a straightforward definition of the force at a given point, once we have decided on the required velocity field (and hence the velocity required at the point in question).

We now turn our attention to the choice of flow field. The implementations to date can be broadly divided into two classes; ghost fluid and numerical boundary layers. In the former, the flow field in the region of interest is extrapolated across the interface in such a way as to impose the desired boundary condition at the interface. This is the method used in the original implementations of Goldstein et al. (1993) and Yusof (1996), as well as in this work. Such an implementation is natural in situations where the fictitious flow produced within the rigid body does not affect the solution and is easily accounted for. This choice has the advantage that the force applied in the fluid region can be zero; that is, the governing equations are unmodified in this region. Additionally, the use of the ghost fluid region allows the effect of, for example, implicit diffusion operators, to be minimized by forcing linear velocity gradients across the interface.

In the latter method, force applied at the interface is numerically smoothed over several grid-points, for numerical stability reasons. As used by Peskin, this is a natural implementation,

since the flow on both sides of the interface is required for the solution. It is possible to use the numerical boundary layer formulation for rigid body problems, as was done by Verzicco et al. (2000), where the discrete-time formulation of Yusof was applied with numerical boundary layers in the fluid side, with exact rigid body fields imposed in the solid.

### 2.3.1 Solution Approach

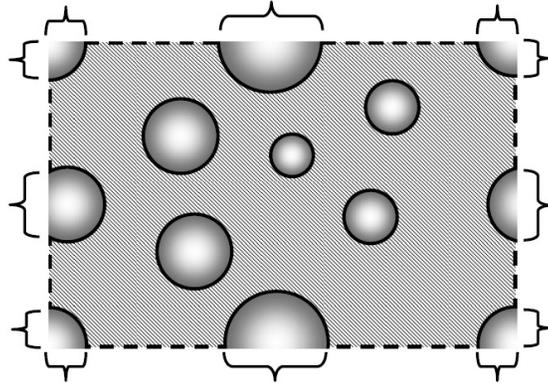


Figure 2.2 Schematic of the computational domain with multiple particles in IBM simulations. Due to periodic boundary conditions, some particles can intersect the domain boundaries. The bounding surfaces of the computational domain, solid-phase, and fluid-phase are denoted by  $\partial\mathcal{V}$ ,  $\partial\mathcal{V}_s$ , and  $\partial\mathcal{V}_f$ , respectively. The total bounding surface of the domain  $\partial\mathcal{V} = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}_f^{\text{ext}}$ , where  $\partial\mathcal{V}_s^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_s$  (shown by curly braces) is the domain surface cut by the solid particles, and  $\partial\mathcal{V}_f^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_f$  (shown by dotted line) is the remaining domain surface. Therefore, the total bounding surface of the solid-phase is  $\partial\mathcal{V}_s = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}_s^{\text{int}}$ , where  $\partial\mathcal{V}_s^{\text{int}}$  (shown by solid lines) is along the actual surface area of the solid particles. Similarly, the total bounding surface of the fluid-phase is  $\partial\mathcal{V}_f = \partial\mathcal{V}_f^{\text{ext}} \cup \partial\mathcal{V}_f^{\text{int}}$ . The hatched lines represent the fluid-phase volume  $\mathcal{V}_f$  and the solid fill represents the solid-phase volume  $\mathcal{V}_s$ , such that the total volume of the computational domain  $\mathcal{V} = \mathcal{V}_s + \mathcal{V}_f$ .

In the immersed boundary method, the mass and momentum equations are solved in the entire domain that includes the interior regions of the solid particles as well. The mass and

momentum conservation equations solved in IBM are

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (2.12)$$

and

$$\rho_f \frac{\partial u_i}{\partial t} + \rho_f S_i = -g_{\text{IBM},i} + \mu_f \frac{\partial^2 u_i}{\partial x_j \partial x_j} + f_{u,i}, \quad (2.13)$$

respectively, where  $\mathbf{g}_{\text{IBM}}$  is the pressure gradient,  $\mathbf{S} = \nabla \cdot (\mathbf{u}\mathbf{u})$  is the convective term in conservative form, and  $\mathbf{u}$  is the instantaneous velocity field. In the above momentum conservation equation,  $\mathbf{f}_u$  is the additional immersed boundary force term that accounts for the presence of solid particles in the fluid-phase by ensuring zero slip and zero penetration boundary conditions (Eq. 2.5) at the particle–fluid interface. The immersed boundary force is computed only at points lying inside the solid particles. At these points, the fluid velocity field is forced in a manner similar to the ghost cell approach used in standard finite-difference/finite-volume based methods. Or more specifically for the case of zero solid particle velocity, the velocity field inside the solid particle at grid points close to the interface is forced to be exact opposite of the fluid velocity field outside the particle. The details of this forcing approach are given in Yusof (1996). In Yusof's original implementation, in addition to the interior points, IB forcing was also computed on the surface of the solid particles that was then interpolated to the neighboring grid nodes - that could include even the grid points in the fluid-phase. This additional forcing leads to contamination of the fluid velocity and pressure fields by the IB forcing, and, therefore, it is not computed anymore. In addition to the forcing the velocity field, the IB forcing term also cancels the remaining terms in the momentum conservation and, at the  $n + 1^{\text{th}}$  time-step, it is given by

$$f_{u,i}^{n+1} = \rho_f \frac{u_i^d - u_i^n}{\Delta t} - \rho_f S_i^n + g_i^n - \mu_f \frac{\partial^2}{\partial x_j \partial x_j} u_i^n, \quad (2.14)$$

where  $u_i^d$  is the desired velocity at that location.

Since the immersed boundary force is a function of both space and time, its effect on the pressure field is accounted by solving a modified pressure Poisson equation given by

$$\frac{\partial g_{\text{IBM},i}}{\partial x_i} = \frac{\partial}{\partial x_i} f_{u,i} - \rho_f \frac{\partial}{\partial x_i} S_i \quad (2.15)$$

which is obtained by taking the divergence of the instantaneous momentum conservation equation (Eq. 2.13).

For the problem of flow past a statistically homogeneous particle assembly, we solve the IBM governing equations by imposing periodic boundary conditions on fluctuating variables that are now defined. The schematic of the computational domain in IBM simulations with multiple particles is shown in Fig. 2.2. Like in the previous section, the bounding surfaces of the computational domain, solid-phase, and fluid-phase are denoted by  $\partial\mathcal{V}$ ,  $\partial\mathcal{V}_s$ , and  $\partial\mathcal{V}_f$ , respectively. Due to periodic boundary conditions, it is possible that some particles intersect the boundaries of the computational domain. Therefore, the total bounding surface of the domain is decomposed as  $\partial\mathcal{V} = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}_f^{\text{ext}}$ , where  $\partial\mathcal{V}_s^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_s$  (shown by curly braces) is the domain surface cut by the solid particles, and  $\partial\mathcal{V}_f^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_f$  (shown by dotted line) is the remaining domain surface. Therefore, the total bounding surface of the solid-phase is  $\partial\mathcal{V}_s = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}^{\text{int}}$ , where  $\partial\mathcal{V}^{\text{int}}$  (shown by solid lines) is along the actual surface area of the solid particles. Similarly, the total bounding surface of the fluid-phase is  $\partial\mathcal{V}_f = \partial\mathcal{V}_f^{\text{ext}} \cup \partial\mathcal{V}^{\text{int}}$ . The hatched lines represent the fluid-phase volume  $\mathcal{V}_f$  and the solid fill represents the solid-phase volume  $\mathcal{V}_s$ , such that the total volume of the computational domain  $\mathcal{V} = \mathcal{V}_s + \mathcal{V}_f$ .

If  $Q(\mathbf{x}, t)$  is any field, then its volume average  $\langle Q \rangle_{\mathcal{V}}$  over the domain volume  $\mathcal{V}$ , referred to as volumetric mean, is defined as:

$$\langle Q \rangle_{\mathcal{V}}(t) = \frac{\int_{\mathcal{V}} Q(\mathbf{x}, t) d\mathcal{V}}{\mathcal{V}}, \quad (2.16)$$

where it is noted that the volumetric mean does not depend on  $\mathbf{x}$ . The fluid-phase  $\langle Q^{(f)} \rangle_{\mathcal{V}}$  and solid-phase  $\langle Q^{(s)} \rangle_{\mathcal{V}}$  volumetric averages are similarly defined. For the statistically homogeneous suspensions, the volumetric mean  $\langle Q \rangle_{\mathcal{V}}$  is an approximation of the true expectation  $\langle Q \rangle$ . In the limit of infinite box-size (i.e.,  $\mathcal{V} \rightarrow \infty$ ), the volumetric mean tends to the true expectation.

From the definition of volumetric mean, the velocity field can be decomposed as the sum of a volumetric mean and a fluctuating component

$$\mathbf{u}(\mathbf{x}, t) = \langle \mathbf{u} \rangle_{\mathcal{V}}(t) + \mathbf{u}'(\mathbf{x}, t), \quad (2.17)$$

and similar decompositions can be written for the non-linear term  $\mathbf{S}$ , pressure gradient  $\mathbf{g}$ , and immersed boundary forcing  $\mathbf{f}_u$  terms. Substituting the above decompositions in the mass (Eq. 2.12) and momentum (Eq. 2.13) conservation equations, followed by volume averaging, yields the mean mass and momentum conservation equations. Since the volumetric means are independent of  $\mathbf{x}$ , mean mass conservation is trivially satisfied. The mean momentum conservation equation becomes

$$\rho_f \frac{\partial \langle u_i \rangle_V}{\partial t} = -\langle g_{\text{IBM},i} \rangle_V + \langle f_{u,i} \rangle_V, \quad (2.18)$$

where it is noted that due to periodic boundary conditions, the volume integrals of convective and diffusive terms are zero.

While mean mass conservation is trivially satisfied, the fluctuating velocity field needs to be divergence free, i.e.,

$$\frac{\partial u_i}{\partial x_i} = 0. \quad (2.19)$$

Subtracting the mean momentum conservation equation 2.18 from the instantaneous momentum conservation equation 2.13 yields the following equation for the conservation of fluctuating momentum:

$$\rho_f \frac{\partial u'_i}{\partial t} + \rho_f S'_i = -g'_i + \mu_f \frac{\partial^2 u_i}{\partial x_j \partial x_j} + f'_{u,i} \quad (2.20)$$

Taking the divergence of the above equation and using equation 2.19 results in the following modified pressure Poisson equation for the fluctuating pressure gradient:

$$\frac{\partial g'_{\text{IBM},i}}{\partial x_i} = \frac{\partial}{\partial x_i} f'_{u,i} - \rho_f \frac{\partial}{\partial x_i} S'_i \quad (2.21)$$

The above conservation equations (Eqs. 2.14, 2.18-2.21) are numerically solved to yield the flow around immersed bodies. The evolution of the mean velocity  $\langle \mathbf{u} \rangle_V$  given by Eq. 2.18 is a function of both the mean IB forcing  $\langle f_{u,i} \rangle_V$  and mean pressure gradient  $\langle \mathbf{g}_{\text{IBM}} \rangle_V$  terms. The mean IB forcing term  $\langle f_{u,i} \rangle_V$  is computed by volume averaging the IB force computed by Eq. 2.14. The estimation of mean pressure gradient  $\langle \mathbf{g}_{\text{IBM}} \rangle_V$  is given in the next section.

### 2.3.2 Estimation of mean pressure gradient $\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}$

The mean pressure gradient should evolve in a way that is consistent with the true expectation given by Eq. 2.10. The derivation for  $\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}$  begins by volume averaging the IBM momentum conservation equation 2.13 over the fluid–phase volume. However, before doing so, a subtle point that arises on volume averaging any quantity over the fluid–phase volume is noted.

Consider any vector field  $\mathbf{Q}(\mathbf{x}, t)$  that can be expressed as a gradient of some scalar field, i.e.  $\mathbf{Q}(\mathbf{x}, t) = \nabla\phi$ . The volume average of  $\mathbf{Q}(\mathbf{x}, t)$  over the fluid–phase volume can be decomposed as volume average over the entire domain volume  $\mathcal{V}$  minus the volume average over the solid–phase volume  $\mathcal{V}_s$  or

$$\int_{\mathcal{V}_f} \nabla\phi(\mathbf{x}, t) \, d\mathbf{x} = \int_{\mathcal{V}} \nabla\phi(\mathbf{x}, t) \, d\mathbf{x} - \int_{\mathcal{V}_s} \nabla\phi(\mathbf{x}, t) \, d\mathbf{x} + \mathbf{R}, \quad (2.22)$$

where  $\mathbf{R}$  is the remainder term that accounts for the jump in  $\phi$  at the particle–fluid interface, and  $d\mathbf{x}$  is an infinitesimal volume. For a continuous  $\phi$  field, there is no jump across the interface and the remainder term will be zero. However, in multiphase flows the shear and normal stresses are discontinuous across the interface and, therefore, the jump condition as implied by  $\mathbf{R}$  should be accounted for<sup>1</sup>. Using the Gauss divergence theorem and noting that  $\partial\mathcal{V}_f = \partial\mathcal{V}_f^{\text{ext}} \cup \partial\mathcal{V}_f^{\text{int}}$  (see Fig. 2.2), the above volume integral over the fluid–phase volume can also be written as

$$\int_{\mathcal{V}_f} \nabla\phi(\mathbf{x}, t) \, d\mathbf{x} = \oint_{\partial\mathcal{V}_f} \phi \mathbf{n}^{(f)} \, dA = \oint_{\partial\mathcal{V}_f^{\text{ext}}} \phi \mathbf{n}^{(\text{ext})} \, dA + \oint_{\partial\mathcal{V}_f^{\text{int}}} \phi^{(f)} \mathbf{n}^{(f)} \, dA, \quad (2.23)$$

where  $dA$  is an infinitesimal area,  $\mathbf{n}^{(f)}$  is the normal vector pointing outward from the interior fluid surface into the solid,  $\mathbf{n}^{(\text{ext})}$  is the normal vector pointing outward from the computational domain. In the second term of the third expression,  $\phi^{(f)}$  implies the value of  $\phi$  field on the fluid side of the particle–fluid interface. Since the  $\phi$  field is continuous along the computational

<sup>1</sup>The forcing approach used in IBM results in continuous stress tensors across the interface and there is no need to account for the jump condition by  $\mathbf{R}$ . However, we prefer to use a more general approach here that is consistent with other methods such as the immersed interface method (Lee and Leveque, 2003; Xu and Wang, 2006)

domain  $\partial\mathcal{V}$  (due to periodic boundary conditions), there is no need to specify  $\phi$  as  $\phi^{(f)}$  in the first term.

The volume integral of  $\mathbf{Q}(\mathbf{x}, t)$  over the solid-phase volume can be similarly written as

$$\int_{\mathcal{V}_s} \nabla \phi(\mathbf{x}, t) \, d\mathbf{x} = \oint_{\partial\mathcal{V}_s} \phi \mathbf{n}^{(s)} \, dA = \oint_{\partial\mathcal{V}_s^{\text{ext}}} \phi \mathbf{n}^{(\text{ext})} \, dA + \oint_{\partial\mathcal{V}_s^{\text{int}}} \phi^{(s)} \mathbf{n}^{(s)} \, dA, \quad (2.24)$$

where  $\mathbf{n}^{(s)} = -\mathbf{n}^{(f)}$  is the normal vector pointing outward from the interior solid surface into the fluid. Substituting the above Eqs. 2.23 and 2.24 into the Eq. 2.22, and noting that  $\partial\mathcal{V} = \mathcal{V}_s^{\text{ext}} \cup \mathcal{V}_f^{\text{ext}}$  (Fig. 2.2), the remainder term  $\mathbf{R}$  becomes

$$\mathbf{R} = \oint_{\partial\mathcal{V}^{\text{int}}} (\phi^{(f)} - \phi^{(s)}) \mathbf{n}^{(f)} \, dA. \quad (2.25)$$

With the above derivation of remainder term, we turn back our attention to the derivation of  $\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}$ . Decomposing the pressure gradient term as  $\mathbf{g}_{\text{IBM}} = \langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}} + \nabla \psi$ , the right hand side of the momentum conservation equation 2.13 can be written as

$$\frac{\partial \tau_{ji}^{\text{IBM}}}{\partial x_j} = -\langle g_{\text{IBM},i} \rangle_{\mathcal{V}} - \frac{\partial \psi}{\partial x_j} + \mu_f \frac{\partial^2 u'_i}{\partial x_j \partial x_j} + f_{u,i} = -\langle g_{\text{IBM},i} \rangle_{\mathcal{V}} + \frac{\partial \tau'_{ji}}{\partial x_j} + f_{u,i}, \quad (2.26)$$

where the velocity field has been expanded using Eq. 2.17.

Integrating the momentum conservation equation 2.13 over the fluid-phase volume  $\mathcal{V}_f$  and using the above expression results in

$$\rho_f \mathcal{V}_f \frac{d \langle u_i^{(f)} \rangle_{\mathcal{V}}}{dt} = -\langle g_{\text{IBM},i} \rangle_{\mathcal{V}} \mathcal{V}_f + \int_{\mathcal{V}_f} \frac{\partial \tau'_{ji}}{\partial x_j} \, d\mathbf{x}, \quad (2.27)$$

where it is noted that the volume average of convective term is zero due to periodic boundary condition along  $\partial\mathcal{V}_f^{\text{ext}}$  and zero penetration boundary condition on the fluid-particle interface  $\partial\mathcal{V}_f^{\text{int}}$ . Since the immersed boundary force is zero in the fluid-phase, its volume average over  $\mathcal{V}_f$  is also zero. If the IB forcing is calculated at the particle-fluid interface and spread to the neighboring grid nodes that could lie in the fluid-phase, then the volume average of IB forcing  $\langle \mathbf{f}_u^{(f)} \rangle_{\mathcal{V}}$  over  $\mathcal{V}_f$  will be non-zero. As a result of this contamination of the fluid pressure and velocity fields by the IB forcing, the IBM simulations will not exactly solve for the physical system that we wish to simulate.

Using Eq. 2.22 and the definition of  $\mathbf{R}$  from Eq. 2.25, Eq. 2.27 becomes

$$\rho_f \mathcal{V}_f \frac{d \langle u_i^{(f)} \rangle_{\mathcal{V}}}{dt} = - \langle g_{\text{IBM},i} \rangle_{\mathcal{V}} \mathcal{V}_f + \int_{\mathcal{V}} \frac{\partial \tau'_{ji}}{\partial x_j} \mathbf{d}\mathbf{x} - \int_{\mathcal{V}_s} \frac{\partial \tau'_{ji}}{\partial x_j} \mathbf{d}\mathbf{x} + \oint_{\partial \mathcal{V}^{\text{int}}} (\tau'_{ji}{}^{(f)} - \tau'_{ji}{}^{(s)}) n_j^{(f)} dA. \quad (2.28)$$

The second term in the above equation is zero because the fluctuating stress tensor  $\tau'_{ji}$  is periodic along  $\partial \mathcal{V}$ . Using the decomposition in Eq. 2.24 for the third term in the above equation results in

$$\rho_f \mathcal{V}_f \frac{d \langle u_i^{(f)} \rangle_{\mathcal{V}}}{dt} = - \langle g_{\text{IBM},i} \rangle_{\mathcal{V}} \mathcal{V}_f - \oint_{\partial \mathcal{V}_s^{\text{ext}}} \tau'_{ji} n_j^{(\text{ext})} dA - \oint_{\partial \mathcal{V}^{\text{int}}} \tau'_{ji} n_j^{(s)} dA + \oint_{\partial \mathcal{V}^{\text{int}}} (\tau'_{ji}{}^{(s)} - \tau'_{ji}{}^{(f)}) n_j^{(s)} dA, \quad (2.29)$$

where  $\mathbf{n}^{(f)} = -\mathbf{n}^{(s)}$  has been substituted in the jump term. The surface integral of  $\tau'_{ji}$  is zero along  $\partial \mathcal{V}_s^{\text{ext}}$  due to periodicity. Noting the cancellation of two other terms, the above equation reduces to

$$\rho_f \mathcal{V}_f \frac{d \langle u_i^{(f)} \rangle_{\mathcal{V}}}{dt} = - \langle g_{\text{IBM},i} \rangle_{\mathcal{V}} \mathcal{V}_f - \oint_{\partial \mathcal{V}^{\text{int}}} \tau'_{ji}{}^{(f)} n_j^{(s)} dA. \quad (2.30)$$

Although the immersed boundary forcing  $\mathbf{f}_u$  ensures zero relative velocity at the particle-fluid interfaces, for periodic boundary conditions we need to ensure that the desired fluid-phase mean velocity will be attained. This is because unlike in inflow/outflow boundary conditions where the flow enters at a specified mass flow rate, there is no such mechanism for periodic boundary conditions. Therefore, in order to attain a desired fluid-phase mean velocity  $\langle \mathbf{u}^{(f)} \rangle_{\mathcal{V}}^d$ , the mean pressure gradient  $\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}$  is advanced in pseudo-time such that at the  $n^{\text{th}}$  time step it is given by

$$- \langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}^n = \rho_f \frac{\langle \mathbf{u}^{(f)} \rangle_{\mathcal{V}}^d - \langle \mathbf{u}^{(f)} \rangle_{\mathcal{V}}^n}{\Delta t} + \frac{1}{(1 - \varepsilon_s) \mathcal{V}} \left\{ - \oint_{\partial \mathcal{V}^{\text{int}}} \psi^n \mathbf{n}^{(s)} dA + \mu_f \oint_{\partial \mathcal{V}^{\text{int}}} \nabla \mathbf{u}^n \cdot \mathbf{n}^{(s)} dA \right\}, \quad (2.31)$$

where all quantities in the integrand are evaluated on the fluid side of the fluid-particle interface, and the superscript  $n$  implies the relevant quantities at the time step. This equation for the volumetrically averaged pressure gradient is obtained by substituting a finite difference approximation for the unsteady term in Eq. 2.30 and expanding  $\tau'_{ji}$  (Eq. 2.26). The first term

on right hand side of the above equation drives the volume-averaged mean fluid velocity to its desired value. The mean pressure gradient  $\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}}$  from the above equation and mean immersed boundary forcing term  $\langle \mathbf{f}_u \rangle_{\mathcal{V}}$  are used to evolve the mean velocity by equation 2.18. For a statistically stationary flow, the equations are evolved in pseudo time until the average quantities reach a steady state, at which point the first term on the right hand side of Eq. 2.31 is negligible, and Eq. 2.31 reduces to the numerical counterpart of Eq. 2.10. This establishes that the resulting numerical solution to the IBM governing equations is a valid numerical solution to steady flow past homogeneous particle assemblies. It is once again noted that the above equivalence holds only when the IB forcing is zero at grid points lying inside the fluid-phase. In IBM implementations where the IB forcing is finite in the fluid-phase (such as the original implementation of Yusof (1996), Uhlmann (2005), etc.), an extra term in the form of fluid-phase volume average of the IB forcing  $\langle \mathbf{f}_u^{(f)} \rangle_{\mathcal{V}}$  will appear in the above expression for mean pressure gradient; thereby resulting in non-equivalence between the desired physical system and the actual simulation.

## 2.4 Simulation Methodology

We now describe how the physical parameters of the problem-mean flow Reynolds number and solid volume fraction-are specified in the simulation. For flow past homogeneous particle assemblies, a Reynolds number based on the magnitude of mean slip velocity between the two phases is defined as

$$\text{Re} = \frac{U_{\text{slip}} (1 - \varepsilon_s) D}{\nu_f}, \quad (2.32)$$

where  $U_{\text{slip}} = |\langle \mathbf{u}^{(f)} \rangle - \langle \mathbf{u}^{(s)} \rangle|$  is the magnitude of the mean slip velocity,  $D$  is the particle diameter, and  $\langle \mathbf{u}^{(f)} \rangle$  and  $\langle \mathbf{u}^{(s)} \rangle$  are the fluid-phase and solid-phase mean velocities, respectively. The objective in direct numerical simulations is to solve the instantaneous mass and momentum conservation equations (Eqs. 2.3 and 2.4) subject to the boundary conditions described earlier, in such a way that the resulting volumetric mean slip velocity corresponds to a desired Reynolds number. This system can be solved in three different ways, namely:

1. *Specified mean pressure gradient  $\langle \mathbf{g} \rangle$* : In this method (Hill et al., 2001a,b) mean pressure

gradient is specified as an input and particle velocities are set to zero. As a result, the mean fluid velocity evolves by Eq. 2.18 and the steady-state solution implies a Reynolds number. One drawback of this approach is that the Reynolds number cannot be set *a priori*.

2. *Specified solid-phase mean velocity*  $\langle \mathbf{u}^{(s)} \rangle$ : In this method (van der Hoef et al., 2005; Beetstra et al., 2007) the simulations are carried out in a laboratory frame of reference wherein the mean velocity  $\langle \mathbf{u} \rangle$  is zero. Therefore, from Eq. 2.7, the desired fluid-phase mean velocity  $\langle \mathbf{u}^{(f)} \rangle = -\frac{\varepsilon_s}{(1-\varepsilon_s)} \langle \mathbf{u}^{(s)} \rangle$ . Substituting this expression for desired fluid-phase mean velocity  $\langle \mathbf{u}^{(f)} \rangle$  in Eq. 2.32 results in an expression for  $|\langle \mathbf{u}^{(s)} \rangle|$  in terms of the Reynolds number and other physical properties. In their simulations, the desired solid-phase mean velocity  $|\langle \mathbf{u}^{(s)} \rangle|$  is attained by specifying equal velocities to all particles.
3. *Specified fluid-phase mean velocity*  $\langle \mathbf{u}^{(f)} \rangle$ : In this method, particles are assigned zero velocity. Therefore, from Eq. 2.32, the desired fluid-phase mean velocity  $\langle \mathbf{u}^{(f)} \rangle$  is known in terms of the input Reynolds number and other physical properties.

The advantage of the second and third methods over the first method is that desired Reynolds number can be specified as an input to the simulation, whereas it is an output in the first method. However, there is no relative advantage in choosing between the second and third methods.

The solid volume fraction  $\varepsilon_s$  together with the ratio of computational box length to particle diameter  $L/D$  determines the number of solid particles in the simulation:

$$N_s = \frac{6\varepsilon_s}{\pi} \left( \frac{L}{D} \right)^3. \quad (2.33)$$

#### 2.4.1 Numerical parameters

The ratio of computational box length to particle diameter  $L/D$  and the number of solid particles  $N_s$  are numerical parameters of the simulation. Their influence on the numerical convergence of the IBM simulations is discussed in the following subsections.

The computational box is discretized using  $M$  grid cells in each direction, and this introduces a grid resolution parameter. The number of grid cells is calculated as

$$M = \frac{L}{\Delta x} = \frac{L}{D} D_m \quad (2.34)$$

where  $L$  is the length of the computational box,  $\Delta x$  is the size of each grid cell, and  $D_m$  is the number of grid cells across the diameter of a solid particle. The solution algorithm is advanced in pseudo-time from specified initial conditions to steady state using a time step  $\Delta t$  that is chosen as the minimum of the convective and viscous time steps by the criteria

$$\Delta t = \text{CFL} \times \min \left\{ \frac{\Delta x}{u_{\max}}, \frac{\Delta x^2 (1 - \varepsilon_s)}{\nu_f} \right\}. \quad (2.35)$$

At the beginning of the simulation  $u_{\max} = |\langle \mathbf{u}^{(f)} \rangle|$ , and as the flow evolves the time step adapts itself to satisfy the above criteria.

#### 2.4.2 Estimation of mean drag from simulations

Direct numerical simulation of flow through a particle assembly using the immersed boundary method results in velocity and pressure fields on a regular Cartesian grid. The drag force on the  $i^{\text{th}}$  particle,  $\mathbf{F}_d^i = m^{(i)} \mathbf{A}^{(i)}$ , is computed by integrating the viscous and pressure forces exerted by the fluid on the particle surface. The average drag force on particles in a homogeneous suspension for  $\mu^{\text{th}}$  realization is computed as

$$\{\mathbf{F}_d\}_{\mathcal{V}}^{\mu} = \frac{1}{N_s} \sum_{i=1}^{N_s} m^{(i)} \mathbf{A}^{(i)} = \frac{1}{N_s} \left\{ -\langle \mathbf{g}_{\text{IBM}} \rangle_{\mathcal{V}} \mathcal{V}_s - \oint_{\partial \mathcal{V}_s} \psi \mathbf{dA} + \mu_f \oint_{\partial \mathcal{V}_s} \nabla \mathbf{u} \cdot \mathbf{dA} \right\}, \quad (2.36)$$

which is obtained by integrating the pressure and viscous fields over the surface of each particle. In the third expression of the above equation, the first term is the force on all particles in the volume due to mean pressure gradient, the second term is the drag force due to fluctuating pressure gradient field, and the third term is the viscous contribution to the drag force. This expression for the drag force is for one realization, and it is then averaged over independent realizations in order to average over different particle configurations corresponding to the same solid volume fraction and pair correlation function. The ensemble-averaged drag is

$$\{\mathbf{F}_d\}_{\mathcal{V}, \mathcal{M}} = \frac{\sum_{\mu=1}^{\mathcal{M}} \{\mathbf{F}_d\}_{\mathcal{V}}^{\mu}}{\mathcal{M}}. \quad (2.37)$$

which converges to the true expectation of the drag force (given by Eqs. 2.2 and 2.11) in the limit  $N_s \mathcal{M} \rightarrow \infty$ . The ensemble-averaged drag force is later reported as a normalized average drag force given by

$$F = \frac{\{\mathbf{F}_d\}_{\mathcal{V}, \mathcal{M}}}{F_{\text{Stokes}}}, \quad (2.38)$$

where  $F_{\text{Stokes}} = 3\pi\mu_f D U_{\text{slip}} (1 - \varepsilon_s)$  is the Stokes drag force.

Each numerical parameter must be chosen to ensure numerically converged, accurate, and physically meaningful results. Spatial and temporal discretization contribute to numerical error in the force on the  $i^{\text{th}}$  particle that scales as  $O(\Delta x^p, \Delta t^q)$ , where  $p$  and  $q$  depend on the order of accuracy of the method and the interpolation schemes at the particle boundary. For steady flow, the numerical error due to spatio-temporal discretization is determined solely by the spatial resolution parameter  $\Delta x/D = 1/D_m$ , which must be sufficiently small to ensure converged results. For the case where the particle positions are chosen to be randomly distributed, on each realization of the flow the computational domain should be chosen large enough so that spatial auto-correlation in the particle force decays to zero. This guarantees that the periodic boundary condition does not introduce artificial effects due to interaction between the periodic images. For a given solid volume fraction  $\varepsilon_s$ , this determines a minimum value of  $N_s = \lceil \varepsilon_s \mathcal{V} \rceil$ . The number of multiple independent simulations  $\mathcal{M}$  is determined by the requirement that the total number of samples  $\sum_{\mu=1}^{\mathcal{M}} N_\mu$  in the estimate for the average force given by Eq. 2.37 is sufficiently large to ensure low statistical error.

Owing to the periodic lattice arrangement of particles in ordered arrays, it is sufficient to solve the flow for just one unit cell (i.e., one particle for the simple cubic (SC) lattice, and four particles for the face-centered cubic (FCC) lattice). For the special case of ordered arrays, since the number of particles is pre-determined, the ratio of computational box length to particle diameter is not an independent numerical parameter. For the ordered arrays, the only numerical parameter is  $D_m$ , which determines the number of grid cells  $M$  required to resolve the flow.

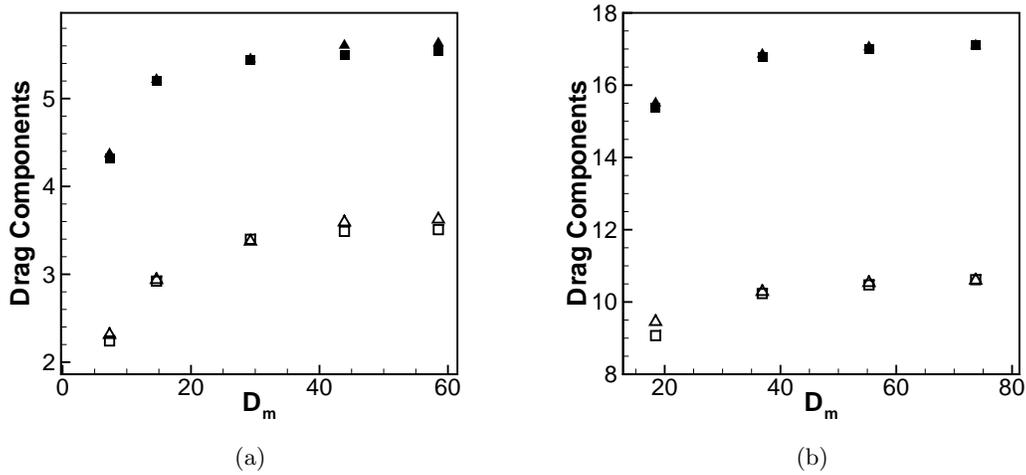


Figure 2.3 Convergence characteristics of drag force due to fluctuating pressure gradient (open symbols) and viscous stresses (filled symbols) for FCC arrays with grid resolution  $D_m$  for two CFL values of 0.2 (squares) and 0.05 (triangles).  $Re = 40$  and volume fraction  $\varepsilon_s$  is equal to 0.2 in (a) and 0.4 in (b).

### 2.4.3 Numerical Convergence

Here we establish that the IBM simulations result in numerically converged solutions. The test case chosen is steady flow past an ordered array of particles in a lattice arrangement, because for this case the only numerical parameter is the grid resolution  $D_m$ . Although we consider steady flows, we also verify that the time step chosen to evolve the flow in pseudo time from a uniform flow initial condition does not change the steady values of drag that we compute using IBM. For an FCC arrangement of particles ( $\varepsilon_s = 0.2$ ,  $Re = 40$ ), Fig. 2.3(a) shows the convergence characteristics of drag forces due to fluctuating pressure gradient (open symbols) and viscous stresses (filled symbols) as a function of grid resolution  $D_m$  for two different values of CFL equal to 0.2 (squares) and 0.05 (triangles). Fig. 2.3(b) shows the same convergence characteristics for a denser FCC arrangement with a solid volume fraction of 0.4 and  $Re = 40$ . In both figures it can be seen that the IBM simulation result does not depend on the time step (CFL). With regard to spatial convergence, the figures show that the resolution requirements increase with increasing volume fraction. While a minimum resolution of  $D_m = 20$  is needed

for converged results at  $\varepsilon_s = 0.2$ , the minimum resolution requirement increases to  $D_m = 30$  for  $\varepsilon_s = 0.4$ . In addition to the dependence of grid resolution on volume fraction, increasing the Reynolds number also requires progressively higher grid resolution. Therefore, for the IBM simulations of ordered arrays that are reported later, higher resolutions are used for the same volume fractions shown in Figures 2.3(a) and 2.3(b), so that the higher Reynolds number cases are also adequately resolved.

When studying grid convergence of a numerical scheme it is sometimes useful to calculate the order of convergence that is implied by the numerical tests. However, the use of a regular Cartesian grid to solve for flow over spheres necessitates interpolation of pressure and viscous stresses from the grid to a finite number of particle surface points. For ordered arrays these interpolation errors cause the steady drag values to exhibit a weak dependence on the location of the particle in the computational box (drag values can differ up to a maximum of 1%). Even for a fixed particle location in the computational box, the interpolation error depends on both the number of particle surface points and the grid resolution. These non-systematic interpolation errors preclude a reliable estimation of the order of convergence of the numerical scheme, which is formally at least second-order. Although the non-systematic interpolation errors prohibit the reliable quantification of spatial order of convergence, if a relative error is defined based on the solution at the finest grid, then a spatial order of convergence in the range 1.5-2 is obtained for the above cases. In other IBM studies (Ikeno and Kajishima, 2007), solution on a highly resolved unstructured grid is taken as a reference to compare the IBM solutions and convergence rates up to second order have been reported.

For the random arrays, in addition to errors arising from finite resolution, errors arise due to statistical fluctuations between different realizations and the box length is also an independent numerical parameter. Ideally, the effect of each numerical parameter on the numerical error should be investigated by varying that parameter while holding the other numerical parameters at fixed values. However, the choice of some numerical parameters must satisfy more than one requirement, and some error contributions are determined by the choice of more than one numerical parameter. Specifically, the choice of  $L/D$  is determined by more

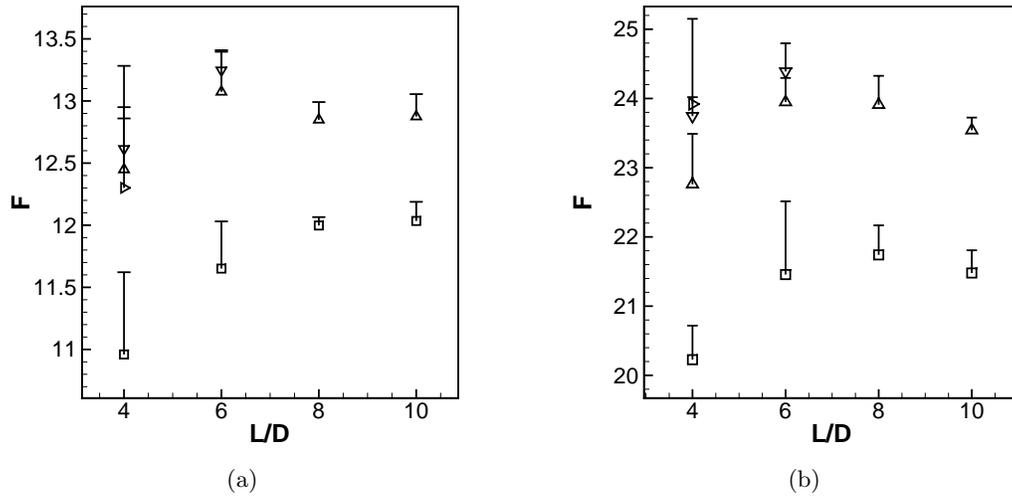


Figure 2.4 Convergence characteristics for random arrays at  $Re = 20$  of the normalized force with box length to particle diameter ratio  $L/D$  for four different values of  $D_m$  equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Solid volume fraction is equal to 0.3 in (a) and 0.4 in (b). Drag values have been averaged over 5 multiple independent simulations. Not all combinations of  $D_m$  and  $L/D$  are shown because with a serial code some combinations exceeded computational memory requirements.

than one requirement (decay of spatial autocorrelation and the need for minimum number of samples in the average force estimate), and both  $L/D$  and the number of multiple independent simulations  $\mathcal{M}$  determine the number of samples in the force estimate. These considerations, as well as computational limitations, did not permit the independent variation of numerical parameters. Therefore, a limited investigation of numerical parameter variation is presented here. To place this in context, we note that to our knowledge this is the most comprehensive study of numerical error and convergence for DNS of gas-solids flow.

While for ordered arrays the box length and number of particles are determined by the volume fraction and type of lattice arrangement (SC/FCC), in random arrays these parameters have to be carefully chosen. If  $L/D$  is too small, then the spatial autocorrelations that are larger than the box size will not be captured and the periodic images will interact. For steady flow past random arrays ( $\varepsilon_s = 0.3$ ,  $Re = 20$ ), Fig. 2.4(a) shows the convergence characteristics

of the normalized force with box length to particle diameter ratio  $L/D$  for four different values of equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Fig. 2.4(b) is the same comparison for the denser random arrays with a volume fraction equal to 0.4. These results show that the drag value depends on  $L/D$  if the simulation is under-resolved, and the effect of grid resolution  $D_m$  is stronger than  $L/D$  for the cases considered here. Once the drag values are at their grid-converged values, there is no statistically significant dependence for  $L > 6D$  in these cases. The simulations of flow past random arrays that are reported later in this work use higher resolutions when the Reynolds number exceeds 100%, as shown in Tab. 2.1.

In summary, these numerical convergence test results show that the IBM simulations yield grid-independent results, and these results are also independent of the choice of time step used to advance the solution in pseudo time, provided the stability criterion is met. The tests for random arrays also show that the grid-converged results do not exhibit a statistically significant dependence on the computational box length for  $L > 6D$ . However, these specific values for the numerical parameters should be treated as tentative because these limited set of tests cannot establish sharp limits on the minimum resolution required, and further numerical testing could refine these limits. A satisfactory number of MIS should ideally be determined by determining the minimum number of samples for a given level of statistical error in the force estimate. However, this quantity is a strong function of  $Re$  and solid volume fraction. In the plots shown above, we have used 5 MIS for all the cases. While this results in a statistical error that is on the order of the other numerical error contributions, further testing is needed to refine this requirement. Clearly, the requirements of minimum  $L/D$ , minimum  $D_m$ , and minimum  $\mathcal{M}$ , together dictate a trade-off for a fixed level of computational work. Of these parameters, our tests reveal that the numerical error in IBM exhibits the highest sensitivity to grid resolution  $D_m$ . These numerical convergence tests provide useful guidelines in the choice of these parameters that approximately balance the error contributions, but further testing is needed for a complete error analysis.

## 2.5 Validation Tests

### 2.5.1 Isolated Sphere

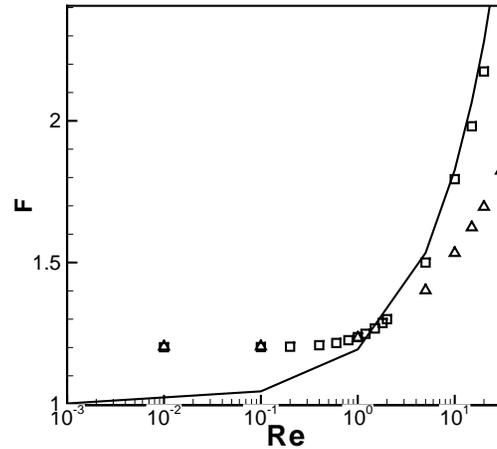


Figure 2.5 Normalized drag force  $F$  in a simple cubic array ( $\varepsilon_s = 4.0 \times 10^{-4}$ ) as a function of Reynolds number and angle  $\theta$  between the mean flow and the  $x$ -axis in the  $(x, y)$  plane. The symbols are from the IBM simulations:  $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ ). The solid line is the drag correlation for an isolated sphere in unbounded medium (Schiller and Naumann, 1933).

The flow over an isolated sphere in an unbounded medium presents itself as the logical validation test for any direct numerical simulation approach to gas-solid flow. However, especially for simulations that use periodic boundary conditions, this turns out to be a difficult validation test. For simulations using periodic boundary conditions, flow through a very dilute simple cubic arrangement is taken as a close approximation to flow over an isolated sphere in an unbounded medium. Since the simple cubic lattice arrangement is not isotropic, it is known (Hill et al., 2001b) that the results for drag can depend on the orientation of the flow with respect to the unit vectors of the lattice for values of Reynolds number beyond the Stokes flow regime. In contrast, there is of course no preferred direction for flow over an isolated

sphere in an unbounded medium.

Fig. 2.5 shows the comparison of the normalized drag force in a simple cubic array ( $\varepsilon_s = 4.0 \times 10^{-4}$ ) as a function of the Reynolds number from IBM simulations with a well-established correlation for an isolated particle in an unbounded medium (Schiller and Naumann, 1933). The drag computed for mean flow oriented at two different angles ( $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ )) with respect to the lattice unit vector is shown to illustrate the dependence on flow angle. For  $Re < 1$  (in the Stokes regime), the normalized drag force is independent of the mean flow angle. However, the drag from IBM simulations is about 20% higher than the established correlation. The drag computed from IBM is within 4% of LBM simulations of dilute SC arrays using periodic boundary conditions. The interactions between the periodic images of the spheres result in higher drag values than an isolated sphere. It is expected that as the volume fraction is further reduced, the numerical predictions will get closer to the drag law in the Stokes limit. The sphere resolution for the simulation shown is equal to 12.8 grid cells. Even more dilute simulations will require larger computational grids.

For  $Re > 1$ , the IBM results are in good agreement with the existing drag law only when the mean flow is directed at an angle of  $\pi/16$  in the  $(x, y)$  plane. This observation is consistent with the earlier LBM (Hill et al., 2001b) simulations where the authors argued that for mean flow angles close to 0 or  $\pi/4$ , the inertial contributions (or pressure gradient contributions) are reduced due to relatively larger wake interactions than for the case of  $\theta = \pi/16$ . The lower inertial contributions result in a lower value for total drag for those flow angles. For  $Re < 1$  the normalized drag force value is independent of the mean flow angle because momentum transport is dominated by viscous diffusion. Since diffusion is symmetric about a sphere, the mean flow angle has no effect on the total drag force in the Stokes regime.

### 2.5.2 Stokes Flow

Several correlations have been proposed in the literature for the drag force in Stokes flow past ordered arrays (SC, FCC, BCC) of spheres. Different analytical and numerical techniques, such as analytical solution to the Stokes equations (Hasimoto, 1959), Galerkin methods (Snyder

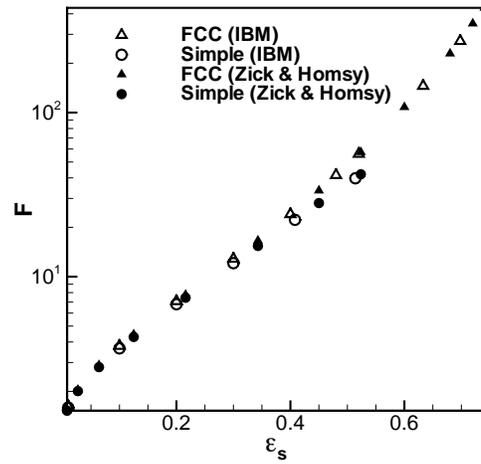


Figure 2.6 Comparison of the normalized drag force  $F$  as a function of the solid volume fraction  $\epsilon_s$  in Stokes flow past SC and FCC arrays from IBM simulations (open symbols) with the results of Zick and Homsy (filled symbols).

and Stewart, 1966; Sorensen and Stewart, 1974a), and the boundary-integral method (Zick and Homsy, 1982), have been used to determine the drag force in Stokes flow past ordered arrays as a function of solid volume fraction. Since Zick and Homsy's results are within 6% of all the other studies, and include all three ordered configurations for the entire range of solid volume fraction, their values are used in Fig. 2.6 as a benchmark to compare with IBM simulations. Fig. 2.6 shows that the IBM simulations are in excellent agreement with reported values from dilute to close-packed limits. The grid resolution in the IBM simulations for the FCC cases is 25.24 and 104 grid points per particle diameter, for the minimum and maximum volume fractions of 0.01 and 0.698 considered, respectively. In the simple cubic cases,  $D_m$  is equal to 40.08 and 149, for the minimum and maximum volume fractions of 0.01 and 0.514, respectively. The validation tests described in this section show that the IBM simulations faithfully reproduce many standard results published in the literature. In cases where there are differences, these are within acceptable limits, and are mostly due to the higher resolution used in the IBM simulations. Having established that the IBM simulations are numerically

convergent and having validated them in standard tests, we now use IBM to study drag in steady flow past ordered and random arrays.

## 2.6 Ordered Arrays

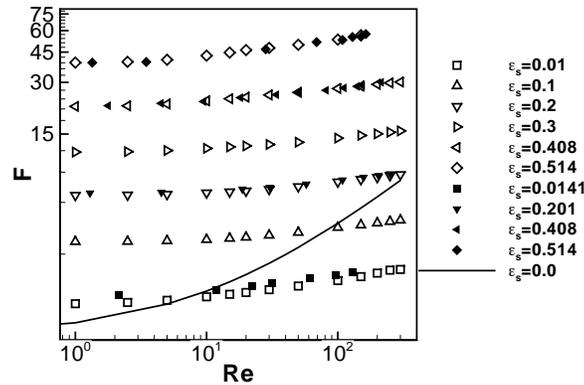


Figure 2.7 Comparison of the normalized drag force  $F$  for SC arrangement obtained from IBM (open symbols) simulations with the LBM simulations (filled symbols) of HKL. The solid line is the Schiller and Naumann drag law for a single particle in an unbounded medium. The flow is directed along the  $x$ - axis.

Ladd and Verberg (2001) and Hill et al. (2001b) have studied steady flow past ordered arrays of particles using LBM simulations. Our purpose in revisiting this problem is two-fold. IBM simulations of flow past ordered arrays serve to further validate the method for cases where we can compare with published data of Hill et al. (2001b). Secondly, we have more comprehensively explored the parameter space defined by  $(Re, \varepsilon_s)$ , especially the low volume fraction region, with higher numerical resolution than reported thus far in the literature. The dilute cases are more computationally demanding, and have therefore not received as much attention. However, the behavior of the drag force in the dilute limit is important because it defines a limiting behavior that drag correlations are typically constrained to satisfy. Our IBM simulations in the dilute regime reveal some new insights into the correct limiting behavior that should be imposed as a constraint on drag correlations. Fig. 2.7 shows the behavior of

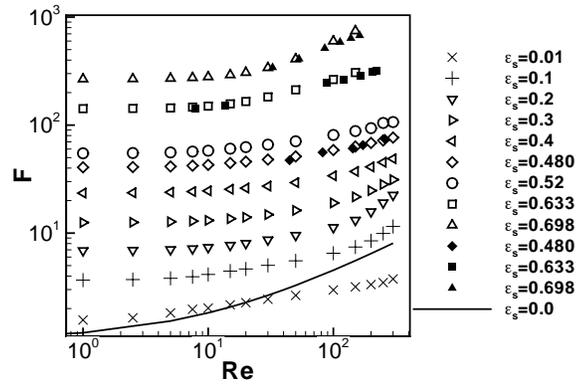


Figure 2.8 Comparison of the normalized drag force  $F$  for FCC arrangement obtained from IBM (open symbols) simulations with the LBM simulations (filled symbols) of HKL. The solid line is the Schiller and Naumann drag law for a single particle in an unbounded medium. The flow is directed along the  $x$ - axis.

the normalized drag force obtained from IBM simulations (open symbols) for steady flow past a SC arrangement of particles as a function of Reynolds number, for volume fractions ranging from very dilute to close-packed limits. Also shown in the same figure is the comparison (wherever the data is available) with the LBM simulations (filled symbols) of HKL. Figure 2.8 shows the same comparison for the FCC arrangement. As both figures show, the IBM and LBM simulations are in excellent agreement. These results serve to further validate the use of IBM for simulation of flow past homogeneous particle assemblies. The solid line in Figures 2.7 and 2.8 is the Schiller and Naumann drag on a single particle in an unbounded medium. Comparison with the single sphere drag law (solid line) reveals that for moderate to high Reynolds numbers, the dilute volume fractions in ordered arrays experience lesser drag than the drag on a single particle. As observed earlier for the dilute SC array (see Fig. 2.5 and its discussion), and also studied comprehensively in HKL, the normalized drag force in ordered arrays is a function of the flow angle. Therefore, in order to avoid the additional parametrization of the problem by flow angle, all the simulations have been performed for the case where the mean flow is directed along the  $x$ - axis. However, as shown in HKL, a change in the flow angle can result

in drag values that differ by as much as 200-300% from the zero flow angle case. The main conclusion to be drawn from these simulations is that the single sphere drag law is not the asymptotic limit of the dilute ordered arrays data. As we shall see in the next section, the same is true for random arrays also, although they do not exhibit the strong dependence on flow angle characteristic of ordered arrays.

## 2.7 Random Arrays

Fixed assemblies of randomly distributed particles are closer to the freely evolving suspension problem that we seek to model than ordered arrays. The random arrays are initialized using a three step algorithm. The details of this algorithm are given in Appendix A and below only the salient features of this algorithm are discussed. First the particles are arranged in a lattice arrangement. For dense volume fractions, this could result in domain length extending beyond the desired box length. In the event of extended box-length, the lattice configuration is shrunk to fit all the particles in the desired volume. This is achieved by applying a spatially decaying force which is symmetric along the center of the extended lattice box length. In order to obtain truly homogeneous particle position distributions (either from after lattice distribution for low volume fractions or from after shrink procedure for denser cases), the particles are assigned a Gaussian velocity distribution and allowed to collide elastically using the soft-sphere collision model (Cundall and Strack, 1978). These final equilibrated and homogeneous particle position distributions are used as input to the IBM simulations.

We have performed IBM simulations with numerical resolutions comparable or higher than those used in HKL and BVK, again with an emphasis on characterizing the dilute limit, which is used to as a limiting case constraint to determine drag law coefficients. Later in this section, the numerical parameters used in the current IBM simulations are compared with those used in the LBM simulations of HKL and BVK. In the following the principal results underlying physical mechanisms and their implications for drag laws are discussed.

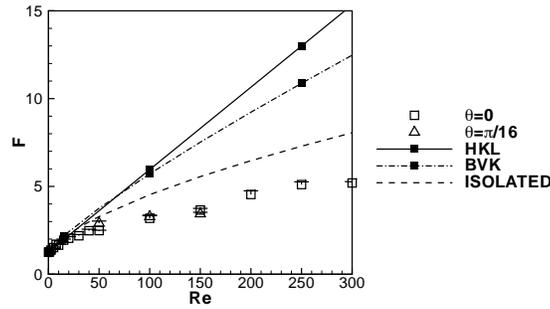


Figure 2.9 Normalized drag force  $F$  Vs Reynolds number for random configuration at solid volume fraction equal to 0.01. Symbols are the IBM simulations, where for squares the flow is directed along the  $x$ - axis, and for triangles the flow is directed at an angle of  $\pi/16$  in the  $x - y$  plane.

### 2.7.1 Dilute Arrays

Fig. 2.9 shows the dependence of normalized drag force  $F$  on the Reynolds number for a random configuration at a dilute volume fraction of 0.01. Symbols are the IBM simulations, with square symbols for the mean flow directed along the  $x$ - axis, and triangles for the mean flow directed at an angle of  $\pi/16$  in the  $x - y$  plane. Solid and dashed-dot lines are the monodisperse drag laws from LBM simulations of HKL and BVK, respectively, and the dashed line is the single sphere drag law of Schiller and Naumann. Comparison of the IBM simulations with existing monodisperse drag laws of HKL and BVK shows an excellent match in the Stokes regime and at low  $Re$ , but differences as high as 100 – 200% in the moderate and high  $Re$  regime. HKL (Hill et al., 2001a) simulated such dilute volume fractions only for the Stokes regime, but due to the coarse resolution of less than 2 lattice nodes for particle diameter they did not simulate higher Reynolds numbers for this volume fraction. BVK, on the other hand, did not simulate any case for  $\varepsilon_s \leq 0.1$ . In HKL, it is noted that due to the approximate approach used to obtain the inertial contribution (denoted by  $F_3$  in their study) to the total drag, their drag law is a good estimate of the actual drag force over the entire range of Reynolds number only for relatively high solid volume fractions. This is a plausible explanation for the departure of IBM simulations from the HKL drag law. The departure of IBM simulations from

BVK's monodisperse drag law is attributed to the incorrect constraint imposed on their drag law to the single-sphere drag correlation at infinite dilution. The BVK drag law assumes that the drag in random homogeneous suspensions at infinite dilution (i.e.,  $\varepsilon_s \rightarrow 0$ ) should tend to the drag on an isolated particle. From both IBM and LBM simulations, it is clear that this assumption does not hold true even at the moderately dilute volume fraction of 0.01.

At low  $Re$ , viscous terms that are local (short range) dominate. Since the viscous forces are short ranged, it is reasonable to expect that at infinite dilution and low  $Re$ , the normalized drag force will approach that of single-sphere drag (i.e.,  $F \rightarrow 1$  as  $\varepsilon_s \rightarrow 0$  and  $Re \rightarrow 0$ ). As the Reynolds number increases, the contribution from inertial terms dominates the viscous effects, and since pressure obeys an elliptic equation these are long range (nonlocal) interactions. For moderate to high Reynolds numbers flow past random arrays, even for fairly dilute solid volume fractions the simulation data do not support the assumption of constraining the drag law to approach that of single-sphere.

Similar to the observations for ordered arrays (Figures 2.7 and 2.8), the drag force on dilute suspensions for moderate to high Reynolds numbers is less than the drag force experienced by an isolated particle in an unbounded medium. However, unlike in ordered arrays, the drag force in random arrays is not dependent on the flow angle due to isotropy of the particle configuration. For ordered arrays, the strong influence of flow angle on the drag force at moderate to high Reynolds numbers is attributed by HKL to the different length scales at which the inertial contributions interact. The distribution of neighbor particles in ordered arrays is anisotropic, and the pair correlation function is sharply peaked at the lattice points. However, in the random particle configurations generated by elastic soft-sphere collisions, the pair correlation is isotropic. Therefore, the drag force is insensitive to flow angle for all Reynolds numbers in random arrays.

### 2.7.2 Moderately Dilute to Dense Arrays

Fig. 2.10 shows the comparison of normalized drag force in random arrays for volume fractions equal to 0.1 and 0.2 obtained from IBM simulations (open symbols) with the existing

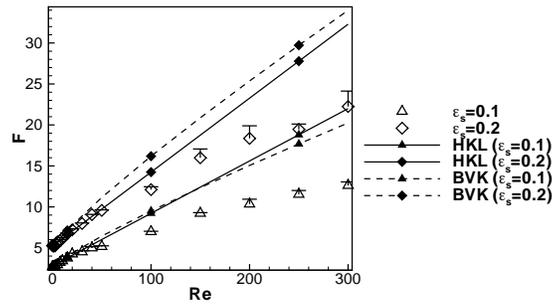


Figure 2.10 Comparison of the normalized drag force  $F$  for random arrays at volume fractions equal to 0.1 and 0.2 from IBM simulations (open symbols) with the monodisperse drag laws of HKL and BVK.

monodisperse drag laws of HKL and BVK. Fig. 2.11 shows the same comparison for volume fractions equal to 0.3 and 0.4. It can be seen that IBM simulations are in excellent agreement with HKL's drag law for  $Re$  up to 100, which is nearly the upper limit of Reynolds number simulated by HKL. The extension of their drag law to higher  $Re$  does not agree well with IBM simulations as the solid volume fraction is reduced. This is attributed to the observation made in HKL that their drag law is a good estimate of the actual drag force over a wide range of Reynolds number only for relatively high solid volume fractions.

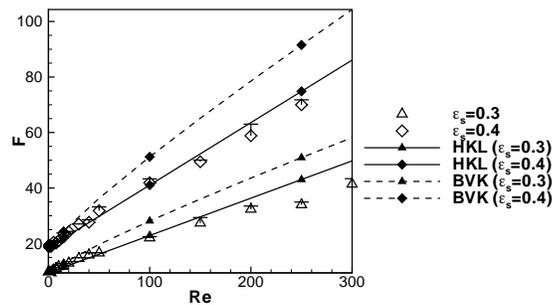


Figure 2.11 Comparison of the normalized drag force  $F$  for random arrays at volume fractions equal to 0.3 and 0.4 from IBM simulations (open symbols) with the monodisperse drag laws of HKL and BVK.

### 2.7.3 Numerical Parameters and Resolution

Choosing the numerical resolution for random arrays or a fixed level of computational work should be based on an optimal combination of box-size and grid resolution. Tab. 2.1 compares the numerical resolutions for different volume fractions used in our IBM simulations with those used in LBM simulations of HKL and BVK. It is noted that not all choices of the numerical parameters for IBM are in the “resolved” range as determined by our limited set of numerical convergence tests. However, as noted earlier, these tests are themselves not comprehensive, and so ultimately the choice of numerical parameters reflects an attempt to balance various contributions to the numerical error. Given the relatively low sensitivity of the mean drag force to  $L/D$  ratio in IBM, we have used values from past LBM simulations as a guideline, choosing higher grid resolution over larger box size in some of our IBM simulations.

For HKL, the numerical resolutions are those used for the maximum Reynolds numbers and are taken from Tab. 2.1 in Hill et al. (2001b). For Stokes flow, HKL have used similar numerical resolution for  $\varepsilon_s \geq 0.1$ . However, for very dilute volume fractions, very coarse resolutions of less than 2 lattice nodes across a particle diameter have been used. In BVK, a constant resolution of 17.5 lattice units across a particle diameter was used for  $\varepsilon_s \leq 0.2$ , and for higher volume fractions, their results were obtained by averaging over two resolutions of 17.5 and 25.5 lattice units. Therefore, in Tab. 2.1, we have used the average value of 21.5 lattice units to report their resolutions for  $\varepsilon_s \geq 0.3$ . In both the studies, random configurations for volume fraction less than 0.1 were not simulated for the entire range of Reynolds numbers and, as a result, there is no numerical resolution comparison for  $\varepsilon_s = 0.01$ . The table shows that the IBM simulations are consistently better resolved in terms of the number of particles, grid resolution, and the box-size. BVK performed greater number of MIS but the scatter in IBM data does not point to a need for such high number of MIS. Therefore, normalized drag values averaged over 5 MIS are reported here.

$\varepsilon_s$	$N_s (\mathcal{M})$	$D_m$	$L/D$
0.01	–	–	–
	–	–	–
0.1	64 (5), 13 (5)	10, 20	15, 9
	16 (5)	9.6	4.38
	54 (20)	17.5	6.6
	80 (5), 41 (5)	20,30	7.5, 6
0.2	16 (5)	17.6	3.47
	54 (20)	17.5	5.2
	161 (5), 34 (5)	20, 40	7.5, 4.5
0.3	16 (5)	17.6	3.06
	54 (20)	21.5	3.07
	71 (5), 26 (5)	30, 50	5, 3.6
0.4	16 (5)	33.6	2.73
	54 (20)	21.5	4.13
	95 (5), 20 (5)	30, 60	5, 3

Table 2.1 Comparison of the numerical resolutions used for random arrays in IBM simulations with the past LBM simulations of HKL and BVK. For each entry, first and second rows correspond, respectively, to the LBM simulations of HKL and BVK, and the third row corresponds to the current IBM simulations. For the IBM simulations, the numbers before and after “,” are, respectively, the resolutions for  $Re \leq 100$  and  $Re > 100$  .

## 2.8 Summary

IBM simulations show an excellent match with the drag correlations proposed by HKL and BVK for low Reynolds number for both dilute and moderately dense random arrays. However, the IBM simulations show a significant departure from these correlations at higher Re, and for dilute cases. The drag law proposed by HKL is stated to be more reliable for all Reynolds numbers only at higher volume fraction. The BVK drag correlation is proposed based on a fit to 5 drag values over a wide range of Reynolds number, and their simulations appear to be susceptible to numerical resolution errors. For a given volume fraction, they used a constant resolution of the particle diameter to simulate Reynolds numbers ranging from 21 to 1000. As the volume fraction is increased, the number of grid/lattice nodes in the gaps between the spheres decrease and a progressively higher grid resolution is required. In the HKL study the particle resolution was increased from 9.6 lattice units per particle diameter for the lowest volume fraction of 0.1 to 41.6 lattice units for the highest volume fraction of 0.641, which is a four-fold increase. However, in the BVK study the particle resolution increased by only a fraction for a wide volume fraction range of 0.1-0.6. The IBM simulations suggest that a more complete parametric study at high resolution could significantly revise these existing drag laws.

## 2.9 Assessment of IBM for drag law formulation

Simulations of steady flow past homogeneous particle assemblies using IBM reveal that fundamentally differing computational approaches to gas-solids flow are in remarkably good agreement for a wide variety of test cases. Overall this is strong evidence of the consistency between different computational approaches to the problem of drag law formulation in gas-solids flow, which is difficult to study through experiment. However, all computational predictions of drag in gas-solids flow are subject to uncertainties arising from numerical error, and should be interpreted as accurate only within 5%. In the following we compare and contrast the IBM approach with LBM, which is a popular computational approach for gas-solids flow.

While IBM solves the continuum Navier-Stokes equations, LBM solves for the discrete one-particle velocity distribution function whose evolution is described by the lattice Boltzmann

equation (He and Luo, 1997). It is useful to think of LBM as a solution to the lattice Boltzmann equation, which is obtained by Hermite–Gauss quadrature of the modeled Boltzmann equation. LBM fundamentally differs from continuum solutions to Navier-Stokes equations like IBM because it directly solves for a discrete form of the velocity distribution function at the molecular level. From the LBM solution the hydrodynamic mean fields such as fluid velocity and pressure can be calculated. Since LBM operations are local in physical space, it avoids solving the elliptic pressure Poisson equation that is needed in incompressible continuum flow solvers. This paves way for efficient parallelization of LBM, which has opened door to solving realistic flow problems (Chen and Doolen, 1998). However there are some issues worth considering when using LBM for gas-solids suspension. The restriction of molecular velocities to discrete values on a lattice is now known to be unnecessary, and even undesirable for many flow problems, especially in multiphase flow (Fox, 2008). Another feature of LBM is that it always results in a compressible flow solution, and as a result the solution of incompressible flow at high Reynolds numbers is challenging. In order to reduce the errors due to compressibility effects at higher Reynolds numbers, the viscosity of the fluid has to be reduced (Ladd, 1994a,b). In the particular implementation of LBM used for gas-solids flow (Hill et al., 2001a; van der Hoef et al., 2005), the collision term appearing in the evolution equation is modeled using a linearized collision operator that allows for multiple relaxation time scales (Ladd, 1994a,b).

When we consider suspension flows, other differences arise between IBM and LBM. In LBM the spherical particle is represented by a stair-step lattice approximation, that is, the surface is represented by a set of lattice sites closest to the input diameter  $D_0$ . Due to this discrete representation of the particle surface, the exact value of the diameter is not known *a priori*. Furthermore, the bounce–back scheme used to implement the no slip boundary condition at the particle–fluid interface shifts the actual boundary layer. Therefore, the drag values in LBM simulations are reported in terms of an effective hydrodynamic diameter  $D_h$ . The hydrodynamic diameter depends on the fluid viscosity as well as the particle size. For every choice of kinematic viscosity and particle diameter  $D_0$ , the hydrodynamic diameter  $D_h$  is obtained by calibrating the LBM simulations against the analytical solution of Hasimoto

(1959) for Stokes flow in dilute SC arrangement of spheres. It is important to note that the volume fractions and drag computations are based on  $D_h$  and not on the input diameter  $D_0$ . The input diameter  $D_0$  is only used to compute distances in the bounce-back scheme to implement boundary conditions at the particle-fluid interface. The calibration requirement in LBM simulations in terms of hydrodynamic diameter is not needed in IBM. It is also interesting to note that the drag on the particle reported using  $D_0$  gives first order convergence whereas drag reported using  $D_h$  results in approximately quadratic convergence (Ladd and Verberg, 2001). However, this convergence rate is not independent of the kinematic viscosity of the fluid. Even though the calibration of hydrodynamic diameter is done for a single sphere, the same calibration is used for simulating dense ordered suspensions in the Stokes regime (Ladd, 1994a,b) as well as random arrays at higher Reynolds number (Hill et al., 2001a; van der Hoef et al., 2005).

LBM is a highly efficient and robust solution methodology for gas-solids flow. Overall, it appears that LBM results for mean drag are relatively insensitive to grid resolution when compared with IBM. However, this insensitivity of the LBM solution to grid resolution should be carefully interpreted because LBM yields stable solutions even when the flow is highly under-resolved. For instance, Beetstra et al. (2007) show that the drag force for a dense random packing of 0.5 at Reynolds number equal to 1049 is insensitive to the grid resolution in the range 10 to 50 lattice units per particle diameter. However, for these grid resolutions it is clear that the boundary layers around the particles cannot be resolved at such a high Reynolds number. Some studies also report greater sensitivity of LBM to grid resolution. For example, in the monodisperse simulations of van der Hoef et al. (2005) at a volume fraction of 0.5 in the Stokes flow regime, a strong dependence of the drag force on the grid resolution and kinematic viscosity is observed. The sensitivity of IBM results to grid resolution has already been discussed, and we find that the IBM results for the surface viscous stress show the correct increasing trend with increasing grid resolution as the velocity gradients are better resolved. If IBM is used to simulate high Reynolds number flows with insufficient resolution (e.g.,  $Re > 500$  with grid resolution in Tab. 2.1), the solution becomes unstable because of the non-dissipative

second-order upwind schemes that have been incorporated for high accuracy. This informs the IBM user that higher grid resolution should be employed to obtain stable and accurate solutions.

From the preceding discussion we can see that IBM has some unique advantages in solving gas-solids flow problems that derive from its solution approach to the continuum Navier-Stokes equations. By virtue of its implementation into structured Cartesian grid solvers, it incurs minimal increase in computational cost with increasing number of particles. To give a rough idea of the order of magnitude of the increase in computational cost, the increase is only about 25% going from 2 particles to 97 particles, but the exact value depends on the Reynolds number and volume fraction. The results presented in this chapter show that IBM yields numerically convergent solutions to important hydrodynamic problems in gas-solids flow, which compare well with many established results in the literature. We also find that this powerful tool is capable of giving additional insight into the important limiting case of steady flow past dilute random arrays. Also a more thorough exploration of the volume fraction-Reynolds number parameter space suggests significant changes to existing drag correlations may be required. In the next section we outline future directions for IBM as a computational method for solving gas-solids flow problems.

## 2.10 Conclusion

IBM is a powerful and efficient computational method for direct numerical simulation of gas-solids flow. This contribution connects the quantities computed from DNS using the IBM approach with the interphase momentum transfer term arising in theoretical approaches to gas-solids flow. This correspondence is described at different levels, starting from the one-particle distribution function and leading naturally to the averaged equation in that approach. An important connection of IBM quantities with two-fluid theory is also established. The numerical convergence of IBM is established and its performance in various validation tests is described. It is shown that IBM simulations reproduce known results for the average drag in Stokes flow past ordered arrays. For random arrays, the IBM results reveal interesting

insights in the dilute limit, and suggest changes to existing drag laws may be required following comprehensive exploration of the Reynolds number-solid volume fraction parameter space. The IBM approach is versatile, and can be extended to include effects of gas-phase turbulence, polydispersity in the size distribution of solid particles, and transport of chemical species and heat due to fluid flow.

### CHAPTER 3. SCALAR TRANSPORT AND HEAT TRANSFER PAST ORDERED AND RANDOM ARRAYS OF MONODISPERSE SPHERES IN STOKES FLOW

This chapter is a manuscript in preparation titled “Scalar transport and heat transfer past ordered and random arrays of monodisperse spheres in Stokes flow regime” authored by R. Garg, M. G. Pai, S. Tenneti, and S. Subramaniam.

The average Stokes Nusselt number  $Nu_0$  in random arrays (defined as the Nusselt number in the limit of creeping flow) has been a subject of much controversy in the past, and correlations differing by orders of magnitude have been proposed. Some correlations predict a Stokes Nusselt number  $Nu_0$  value of less than 2 (which is the Stokes Nusselt number value for an isolated particle) and tending to zero (Kunii and Smith, 1961; Littman et al., 1968; Cybulski et al., 1975), while others predicted values as high as 10 (Gunn and Desouza, 1974) and 13 (Pfeffer and Happel, 1964). In this work, we perform direct numerical simulation (DNS) of passive scalar transport in flow past stationary simple cubic (SC), face-centered cubic (FCC), and random arrays of spheres in the Stokes flow regime. The direct forcing immersed boundary method (IBM) of Yusof (1996), which has been successfully used in the last chapter to simulate the flow past particle configurations up to the close packing limit, is extended to solve for passive scalar transport. The ensemble average scalar transport equations are derived, and the resulting unclosed interphase heat transfer term is related to the average Nusselt number. From the scalar IBM simulation, the average Stokes Nusselt number  $Nu_0$  is obtained as a function of solid volume fraction  $\varepsilon_s$  for SC, FCC, and random arrays. Comparison of  $Nu_0$  from our simulations with the widely used heat and mass transfer correlation of Gunn (1978) for fixed and fluidized beds shows that Gunn’s correlation always underpredicts  $Nu_0$  for all values

of solid volume fractions and differs by as much as 300% for the highest solid volume fraction of 0.5 that is considered in this study. A correlation for  $Nu_0$  in random arrays corresponding to the gas–solid systems (Prandtl number  $Pr = 0.7$ ) as a function of the solid volume fraction up to  $\varepsilon_s = 0.5$  is proposed as

$$Nu_0(\varepsilon_s) = 2 + \frac{1}{(1 - \varepsilon_s)^3} \left( 10.35\varepsilon_s + 5.51\varepsilon_s^2 - 18.16\varepsilon_s^{3/2} + 1.63\varepsilon_s^{1/3} \right).$$

From investigation of the local Nusselt number along particle surfaces in ordered arrays, it is found that (in the Stokes flow regime) the maxima and minima of local Nusselt number occur in the regions of widest and narrowest flow channels, respectively. This observation motivates the study of second–order effects due to inter–particle distance on the average Stokes Nusselt number in random arrays. Average inter–particle distance can be quantified by hard–core distance  $h_c$ , which is defined as the minimum distance between the centers of any two particles. In order to generate random arrays having the same volume fraction and number density but differing in hard–core distance, a three–step random particle configuration generation algorithm is developed that can be used up to  $\varepsilon_s = 0.5$ . Scalar IBM simulations of such random configurations differing in hard–core distance reveal that the Stokes Nusselt number increases as a result of increasing hard–core distance. The dependence on hard–core distance increases with solid volume fraction  $\varepsilon_s$  and vanishes in the limit of infinite dilution.

### 3.1 Introduction

Gas–solid flows are ubiquitous in nature and are encountered in many industrial applications. For example, gas–solid fluidized beds are used in a variety of industries, such as food, power–generation, metallurgical, and pharmaceutical. There is also a renewed interest in studying these flows in the context of biomass energy generation and chemical looping combustion (Shen et al., 2009), which are examples of emerging technologies for environmentally friendly energy generation. One of the challenges in the development of these technologies is the design and scale-up of the components involving particle-laden flow. Fluidized beds and pneumatic transport lines where particle–laden flows are usually encountered are notoriously hard to design and scale up. CFD (Syamlal et al., 1993; Kashiwa and Gaffney, 2003; Sun

et al., 2007) simulations of such systems provide an efficient means to optimize design as the experiments are often costly and also time consuming. Since the averaged equations governing mass, momentum, and energy that are solved in CFD simulations are obtained by statistical averaging procedure (Anderson and Jackson, 1967a; Drew and Passman, 1998) the average interaction terms corresponding to mass, momentum, and energy exchange between different phases are modeled through empirical correlations from experimental, theoretical, and numerical studies. Therefore, the accuracy of such CFD simulations is only as good as that of the correlations.

This study focuses on heat and mass transfer in fluidized and fixed beds which are generally quantified by correlations for average Nusselt  $Nu$  number and average Sherwood number, respectively. The average Nusselt number in fluidized and fixed beds is a function of mean flow Reynolds number  $Re$ , solid volume fraction  $\varepsilon_s$ , and Prandtl  $Pr$  number. Similarly, the average Sherwood number by heat and mass transfer is analog is dependent on mean flow Reynolds number, solid volume fraction, and Schmidt number. In this study, the attention is restricted to the special case of gas-solid flows for which the typical value of Prandtl number is equal to 0.7. Also, the study is restricted to Stokes/creeping flow regime, i.e.,  $Re \rightarrow 0$ .

A review of the literature reveals wide disparity in the values for average Stokes Nusselt number  $Nu_0$  – defined as the average Nusselt number in the Stokes flow regime. Most correlations are developed by combining a few point-wise measurements with a model for heat transfer in gas-solid flow. Experimental measurement of heat transfer in gas-solid flow is challenging because of limited optical access. It is also difficult to isolate the errors arising from modeling assumptions from the experimental measurement errors. Therefore, we employ a ‘true’ direct numerical simulation (DNS) where we solve the extended Navier–Stokes equations with exact boundary condition’s imposed on each particle’s surface. The ‘true’ DNS approach is different from point-particle DNS that have been widely reported in the literature. Since point-particle DNS itself uses a model for the interphase interactions, it cannot be used to develop models. However, in ‘true’ DNS, the volume occupied by each particle is represented by finite number of grid points and exact boundary conditions (no slip, and also no penetration for imperme-

able surfaces) are satisfied at the fluid–particle interface. True DNS is needed if one desires model free simulations from which flow physics can be referred, such as Lattice Boltzmann simulations which have recently been used to propose correlations for drag force on monodisperse (Hill et al., 2001b) and bidisperse (Beetstra et al., 2007) assemblies of homogeneously distributed stationary particles.

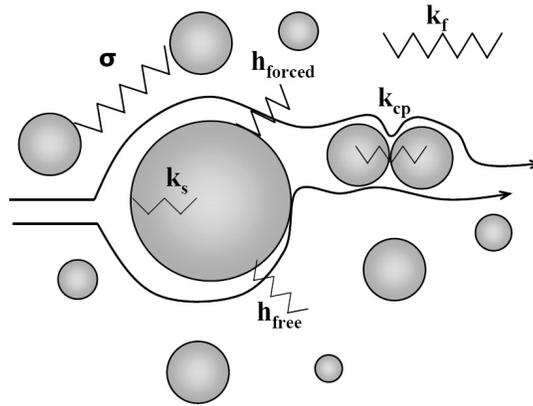


Figure 3.1 Schematic of various heat transfer mechanisms in suspensions.

The drag force on particles in a suspension is the sum of forces resulting from viscous and pressure stress tensors. Besides, the mean drag force acting on a suspension can easily be related to mean pressure gradient across the column (in experiments), or the computational box (in numerical simulations). However, unlike the momentum transfer, heat transfer in suspensions is a complicated process as the heat can be transferred due to multiple mechanisms. The schematic in Fig. 3.1 shows through resistances various mechanisms of inter and intra phase heat transfer in a suspension. As the schematic shows, heat can be transferred through radiative exchanges ( $\sigma$ ) between particles, free ( $h_{free}$ ) and forced ( $h_{forced}$ ) convection at the fluid–solid interphases, and simple conduction in fluid ( $k_f$ ) and solid ( $k_s$ ) phases. Also shown is the heat transfer as a result of conduction ( $k_{cp}$ ) between two touching particles.

The large discrepancies in existing heat transfer correlations is now discussed in some detail. There is a plethora of experimental literature spanning over the last 7 decades (see Wakao

and Kaguei (1982) for a comprehensive review) on heat and mass transfer in packed gas–solid and liquid–solid fixed–bed reactors. Various experimental techniques, such as axial heat conduction in beds (Kunii and Smith, 1961), step response (Handley and Hegg, 1968), frequency response (Gunn and Desouza, 1974; Littman et al., 1968), and shot response (Shen et al., 1981; Wakao et al., 1977), have been used to study heat and mass transfer in fixed–beds. These experimental studies have reported values for the average Stokes Nusselt number in packed beds ( $\varepsilon_s \approx 0.6$ ) that differ by orders of magnitude. For an isolated particle ( $\varepsilon_s \rightarrow 0$ ) in an unbounded medium, average Stokes Nusselt number value of 2 is theoretically known. However, experimental studies on packed beds have reported values for average Stokes Nusselt number on both sides of isolated particle Nusselt number value. For example, some experimental studies on packed beds suggested anomalous average Stokes Nusselt number values less than 2 and tending to zero (Kunii and Smith, 1961; Littman et al., 1968; Cybulski et al., 1975), while others reported values greater than 2 and as high as 13 (Pfeffer and Happel, 1964). In some cases, studies using the same experimental technique have drawn divergent conclusions. For example, although both Gunn and Desouza (1974) and Littman et al. (1968) used the frequency response technique, while Gunn and Desouza (1974) reported average Stokes Nusselt number greater than 2 and tending to 10<sup>1</sup>, Littman et al. (1968) reported average Stokes Nusselt number approaching zero.

These large differences in the average Stokes Nusselt number could be due to the different models used in these studies to infer the gas–particle heat transfer coefficient. A very detailed analysis of the three primary models used in various experimental studies is given in Wakao and Kaguei (1982), and only the key points and fundamental equations of each model are given below:

1. *Schumann Model (Schumann, 1929)*: It is the simplest of the three models. Ideal plug flow is assumed for the fluid–phase and no diffusion is considered in both the fluid and solid phases, i.e., zero temperature gradient within the particle. Under these assumptions,

<sup>1</sup>Due to the low sensitivity to frequency response measurement technique for  $Re < 1$ , Gunn and Desouza (1974) measurements were limited to  $Re = 1$ , based on which they predicted a limiting average Nusselt number value of 10 for packed beds.

the fundamental equations used to model heat transfer in the fluid and solid phases, respectively, are

$$\frac{\partial T_f}{\partial t} = -U \frac{\partial T_f}{\partial x} - \frac{h_P \sigma_A}{(1 - \varepsilon_s) C_{pf} \rho_f} (T_f - T_s), \quad (3.1)$$

and

$$\varepsilon_s \frac{\partial T_s}{\partial t} = \frac{h_P \sigma_A}{C_{ps} \rho_s} (T_f - T_s). \quad (3.2)$$

In the above equations, the subscripts F and S are, respectively, for the fluid and solid phases.  $T$  is the temperature,  $C$  is the specific heat,  $k$  is the thermal conductivity,  $\varepsilon_s$  is the solid volume fraction,  $\sigma_A$  is the particle surface area per unit volume or the interfacial area density, and  $h_P$  is the particle-to-fluid heat transfer coefficient.

2. *Continuous–Solid (C–S) phase model*: In this model, proposed by Littman et al. (1968), in addition to assuming an ideal plug flow in the fluid–phase, the solid–phase is also assumed to be continuous. Additionally, axial conduction is also considered in both phases. Therefore, the fundamental equations used to model heat transfer in the fluid and solid phases, respectively, are

$$\frac{\partial T_f}{\partial t} = \frac{k_{ef}}{(1 - \varepsilon_s) C_{pf} \rho_f} \frac{\partial^2 T_f}{\partial x^2} - U \frac{\partial T_f}{\partial x} - \frac{h_P \sigma_A}{(1 - \varepsilon_s) C_{pf} \rho_f} (T_f - T_s), \quad (3.3)$$

and

$$\varepsilon_s \frac{\partial T_s}{\partial t} = \frac{k_{es}}{C_{ps} \rho_s} \frac{\partial^2 T_s}{\partial x^2} + \frac{h_P \sigma_A}{C_{ps} \rho_s} (T_f - T_s). \quad (3.4)$$

In the above equations,  $k_{ef}$  and  $k_{es}$  are the effective thermal conductivities of the fluid and solid phases, respectively.

3. *Dispersion–Concentric (D–C) model*: In this method, the fluid is assumed to be in the dispersed plug flow mode and a radially symmetric or concentric temperature profile is assumed inside each particle. For this model, the fundamental equations for heat transfer in the fluid and solid phases, respectively, are

$$\frac{\partial T_f}{\partial t} = \alpha_{ax} \frac{\partial^2 T_f}{\partial x^2} - U \frac{\partial T_f}{\partial x} - \frac{\sigma_A}{(1 - \varepsilon_s) C_{pf} \rho_f} k_s \left( \frac{\partial T_s}{\partial r} \right)_R, \quad (3.5)$$

and

$$\frac{\partial T_s}{\partial t} = \frac{k_s}{C_{ps} \rho_s} \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial T_s}{\partial r} \right)_R, \quad (3.6)$$

with

$$k_s \frac{\partial T_s}{\partial r} = h_P (T_f - T_s) \quad \text{at } r = R. \quad (3.7)$$

In the above equations,  $\alpha_{ax}$  is the axial fluid thermal diffusion coefficient. In the original D-C model, by analogy with mass diffusion, the following function form for  $\alpha_{ax}$  is chosen:

$$\begin{aligned} \alpha_{ax}^{\text{orig}} &= (0.6 \sim 0.8)\alpha_f && \text{for } \text{Re} < 1, \\ &= (0.6 \sim 0.8)\alpha_f + 0.5D_P U && \text{for } \text{Re} < 5, \end{aligned} \quad (3.8)$$

where  $\alpha_f = k_f/\rho_f C_{pf}$  is the thermal diffusion coefficient of the fluid-phase.

Even though the three models differ in their treatment of heat transport in each phase, all of them lead to anomalous values of less than 2 for the average Stokes Nusselt number in packed beds. Each study proposed a different mechanism to explain this anomaly (see Wakao et al. (1978) and Wakao and Kaguei (1982) for various proposed mechanisms). However, Gunn and Desouza (1974) obtained a limiting average Nusselt number value of 10 at  $\text{Re} = 1$  for packed-bed from frequency response experiments, which contradicts findings from earlier experimental studies. In their experiments they observed that at low  $\text{Re}$  ( $\text{Re} < 10$ ), heat transfer was dominated by axial diffusion, where axial diffusion is defined as the component of the conductive flux aligned with the mean flow direction. Based on findings of Gunn and Desouza (1974), Wakao et al. (1978) re-evaluated the data from all the experiments that had considered axial diffusion by using the D-C model. Based on analogy with mass transfer, Wakao et al. (1978) suggested the following form for the average Nusselt number in packed beds:

$$\text{Nu} = 2 + 1.1 \text{Pr}^{1/3} \text{Re}^{0.6}. \quad (3.9)$$

It can be seen that this correlation yields an average Stokes Nusselt number of exactly 2 for a packed bed. Wakao et al. (1978) proposed this correlation because of the lack of confidence in experimental data at low  $\text{Re}$  even though enough experimental studies suggested higher values for the average Stokes Nusselt number. For example, the frequency response experiment of Gunn and Desouza (1974) predicted an average Stokes Nusselt number value of 10. Sorensen and Stewart (1974b) studied the creeping flow through simple cubic arrangement of spheres and

obtained an average Stokes Nusselt number of about 4. Using the free surface model, Pfeffer and Happel (1964) obtained an average Stokes Nusselt number value of about 13 as the Reynolds number dropped to zero in a bed with a solid volume fraction  $\varepsilon_s$  of 0.6.

In a subsequent study, (Gunn, 1978) used a stochastic model to describe flow sections in random-packed beds in terms of well-defined geometries under well-defined flow conditions. Using analytically known results for convective and diffusive heat transport for such well-defined geometries under well-defined conditions, he related the convective and diffusive heat transport in random-packed beds to these geometries through a simple stochastic model. Using this statistical representation, Gunn argued that the inclusion of axial diffusion does not alter the asymptotic value of the average Stokes Nusselt number. However, this conclusion contradicts his earlier observation concerning the importance of axial diffusion in low Reynolds number regime in (Gunn and Desouza, 1974). Based on this stochastic model for low Re, and using the experimental data for higher Reynolds numbers, Gunn (1978) proposed a single correlation for particle-to-fluid heat and mass transfer in both fixed and fluidized beds of the form

$$\text{Nu} = (7 - 10\varepsilon_b + 5\varepsilon_b^2) \left(1 + 0.7\text{Re}^{0.2}\text{Pr}^{1/3}\right) + (1.33 - 2.4\varepsilon_b + 1.2\varepsilon_b^2) \text{Re}^{0.7}\text{Pr}^{1/3}, \quad (3.10)$$

where  $\varepsilon_b = 1 - \varepsilon_s$  is the bed porosity. This expression is valid in the bed porosity range  $0.35 \leq \varepsilon_b \leq 1.0$  and for  $\text{Re} \leq 10^5$ . For the case of mass transfer, the average Sherwood number is to substituted for the average Nusselt number Nu, and the Schmidt number for Prandtl number in this correlation. This correlation has been widely used in the CFD simulations of two-phase flows (Syamlal et al., 1993) to simulate heat and mass transfer in both gas-solid and liquid-solid flows.

It is clear from this brief review of experimental studies that there is no consensus on what the average Stokes Nusselt number behavior is a function of solid volume fraction  $\varepsilon_s$ . In this work, heat transfer is studied in fixed beds by simulating steady Stokes flow past homogeneous arrays of spherical particles. The direct-forcing immersed boundary method (IBM) originally developed by Yusof (1996) for simulating flow past assembly of particles, and successfully extended to solve steady flow past homogeneous arrays of spherical particles in the previous

chapter, is extended to solve for passive scalar transport past homogeneous particle assemblies.

The Nusselt number in fixed and fluidized beds is a function of solid volume fraction  $\varepsilon_s$ , Reynolds number  $Re$ , and Prandtl number  $Pr$ . In this study, we restrict ourselves to gas–solid systems where the typical value of Prandtl number is equal to 0.7. For the same volume fraction and number density, it is possible to have random configurations with different hard–core distances  $h_c$ , which is defined as the minimum distance between the centers of any two particles. The second–order effect of hard–core distance on the average Stokes Nusselt number is studied by simulating random arrays, having same solid volume fraction and number density, for different values of hard–core distance.

### 3.2 Governing Equations

The governing equations for mass and momentum conservation in the fluid–phase are given by Eqs. 2.3 and 2.4. The zero slip and zero penetration boundary conditions at the fluid–particle interface is given by Eq. 2.5. For scalar transport, the discussion below is motivated as a heat transfer problem. However, due to the heat–mass transfer analogy, the governing equations below hold for the mass transfer as well. The conservation equation for the scalar field  $\phi$  in the fluid–phase is

$$\rho_f C_{pf} \frac{\partial \phi}{\partial t} + \rho_f C_{pf} \frac{\partial u_j \phi}{\partial x_j} = -\frac{\partial q_j}{\partial x_j} + S_\phi, \quad (3.11)$$

where  $\mathbf{q} = -k_f \nabla \phi$  is the conductive heat flux in the fluid–phase,  $\rho_f$ ,  $C_{pf}$ , and  $k_f$  are the thermodynamic density, specific heat, and thermal conductivity of the fluid–phase, respectively. Since we are interested in homogeneous suspensions, an additional sink/source term  $S_\phi$  has been added to the above equation that adds or removes heat from the fluid–phase at a rate at which it is being drawn or added by the solid particles. Later in this section it is shown that the source/sink term can be used to generate steady–state solution to scalar transport in statistically homogeneous particle assemblies with periodic boundary conditions.

For scalar transport, the discussion below is motivated as a heat transfer problem. However, due to the heat–mass transfer analogy, the governing equations below hold for mass transfer

as well. The conservation equation for a scalar field  $\phi$  in the fluid-phase is

$$\rho_f C_{\text{pf}} \frac{\partial \phi}{\partial t} + \rho_f C_{\text{pf}} \frac{\partial u_j \phi}{\partial x_j} = -\frac{\partial q_j}{\partial x_j} + S_\phi, \quad (3.12)$$

where  $\mathbf{q} = -k_f \nabla \phi$  is the conductive heat flux in the fluid-phase,  $\rho_f$ ,  $C_{\text{pf}}$ , and  $k_f$  are the thermodynamic density, specific heat, and thermal conductivity of the fluid-phase, respectively. A volumetric sink/source is represented by the term  $S_\phi$ . Later it is shown that this sink/source term can be used to generate a nontrivial steady-state solution to scalar transport in statistically homogeneous particle assemblies.

The scalar field in the solid phase also evolves by a similar equation (except for zero convection term inside the solid). However, in this study we neglect scalar gradients inside the solid particles and assume that all particle surfaces are held at a constant surface value of  $\phi_s$  (i.e., isothermal boundary condition). Furthermore, the feedback of scalar transport on the hydrodynamic fields due to density variation (free convection effects) is not considered in this study. These assumptions limit the gas-solid systems to which our simulation results apply. The parameter ranges for both heat and mass transfer in which these results apply are discussed in Sec. 3.2.1. Under the assumption of isothermal boundary condition for particle surface temperature, the following boundary condition must be satisfied by the scalar field at the fluid-particle interface

$$\phi_f = \phi_s \quad \text{on} \quad \partial \mathcal{V}^{\text{int}}. \quad (3.13)$$

These are the equations that are solved in a ‘true’ DNS approach to gas-solid flow. Since the objective of this study is to provide better closure models for the interphase transfer of heat/mass between the phases, we now present the ensemble averaged equations for gas-solid flow. In this way we identify the unclosed terms in the averaged equations and also verify the correspondence between the volume averaged quantities computed from our scalar IBM simulations in Sec. 3.3 and their counterparts in the ensemble averaged equations. A similar comparison for momentum conservation is derived in previous chapter using the Eulerian two-fluid theory (Drew, 1983; Pai and Subramaniam, 2008). Here the averaged equation corresponding to the scalar transport equation 3.12 is derived. Using the definitions of phasic and mixture means (Eq. 2.6 and 2.7), the mean scalar transport equation (Drew, 1983) in

the fluid–phase obtained by multiplying the scalar transport equation 3.12 by the fluid–phase indicator function  $I_f$  is

$$\begin{aligned} \frac{\partial}{\partial t} \left\{ \rho_f \varepsilon_f C_{\text{pf}} \langle \phi^{(f)} \rangle \right\} + \frac{\partial}{\partial x_j} \left\{ \rho_f \varepsilon_f C_{\text{pf}} \langle u_j^{(f)} \rangle \langle \phi^{(f)} \rangle \right\} = \frac{\partial}{\partial x_j} \left\{ \rho_f C_{\text{pf}} \langle I_f u_j''^{(f)} \phi''^{(f)} \rangle \right\} \\ - \left\langle I_f \frac{\partial q_j}{\partial x_j} \right\rangle + \langle S_\phi^{(f)} \rangle \varepsilon_f, \end{aligned} \quad (3.14)$$

where  $u_j''^{(f)} = u_j - \langle u_j^{(f)} \rangle$  and  $\phi''^{(f)} = \phi - \langle \phi^{(f)} \rangle$  are the fluctuating components of the fluid velocity and scalar fields. The terms on the right hand side are the scalar flux transport, divergence of the conductive flux, and source term (all in the fluid–phase). Using the product rule and commuting the derivative and averaging operators, the second term on the right hand side can be expanded as  $\frac{\partial}{\partial x_j} \langle I_f q_j \rangle - \left\langle \frac{\partial I_f}{\partial x_j} q_j \right\rangle$ , where  $\left\langle \frac{\partial I_f}{\partial x_j} q_j \right\rangle$  is the interphase heat transfer term. The gradient of the indicator function  $\nabla I_f$  in this interphase heat transfer term can be expressed as  $-n_j^{(f)} \delta(\mathbf{x} - \mathbf{x}^{(I)})$  (Drew, 1983), where  $n_j^{(f)}$  is the unit normal vector pointing outward from the fluid surface into the particle, and  $\delta(\mathbf{x} - \mathbf{x}^{(I)})$  is a Dirac-delta function concentrated at the fluid–particle interface  $\mathbf{x}^{(I)}$ .

For a statistically homogeneous suspension at steady state (statistically stationary flow), the average quantities do not depend on  $\mathbf{x}$  or  $t$ , and the unsteady and convective terms on the left hand side of Eq. 3.12 do not contribute. Writing the remaining terms shows that the fluid–phase mean of sink/source term balances the average heat flux on the solid particles:

$$\varepsilon_f \langle S_\phi^{(f)} \rangle = - \left\langle q_j n_j^{(s)} \delta(\mathbf{x} - \mathbf{x}^{(I)}) \right\rangle, \quad (3.15)$$

where  $n_j^{(s)} = -n_j^{(f)}$  is the unit normal vector pointing outward from the particle surface into the fluid, and the flux  $\mathbf{q}$  is evaluated on the fluid side of the interface. The other unclosed transport term due to scalar flux (first term on the right hand side of Eq. 3.14) vanishes in the statistical homogeneous case, and is therefore not part of this study. If the sink/source term  $S_\phi$  term was not added to the scalar transport equation 3.12, then a statistically homogeneous suspension would result in zero heat transfer at steady–state. This is because in the absence of sink/source term, the fluid phase temperature field at long time will eventually equilibrate to the solid surface temperature, resulting in zero heat flux at the fluid–particle interfaces.

Similar to the interphase momentum transfer term that appears as an unclosed term on averaging the momentum conservation equation,  $\langle q_j n_j^{(s)} \delta(\mathbf{x} - \mathbf{x}^{(I)}) \rangle$  is an unclosed term describing the interphase heat transfer. In averaged equation solvers, such as MFIX (Syamlal et al., 1993), it is usually modeled as

$$\langle q_j n_j^{(s)} \delta(\mathbf{x} - \mathbf{x}^{(I)}) \rangle = \frac{6\varepsilon_s h_{fs}}{D} \left( \langle \phi^{(s)} \rangle - \langle \phi^{(f)} \rangle \right), \quad (3.16)$$

where  $h_{fs}$  is the average interphase heat transfer coefficient. The factor  $\frac{6\varepsilon_s}{D}$  is the interfacial area density defined as the ratio of the total solid surface area  $\partial\mathcal{V}_s$  to the domain volume  $\mathcal{V}$ . If the expectation of average Nusselt number  $\text{Nu}$  is defined as

$$\langle \text{Nu} \rangle = \frac{h_{fs} D}{k_f}, \quad (3.17)$$

then from Eqs. 3.15 and 3.16,  $\langle \text{Nu} \rangle$  becomes

$$\langle \text{Nu} \rangle = - \frac{\langle S_\phi^{(f)} \rangle}{k_f (\langle \phi^{(s)} \rangle - \langle \phi^{(f)} \rangle)} \frac{1 - \varepsilon_s}{\varepsilon_s} \frac{D^2}{6}, \quad (3.18)$$

The expression for  $\langle \text{Nu} \rangle$  tells us that the average interphase heat transfer can be inferred indirectly from the volumetric source term statistically homogeneous and stationary problem.

In the next section we compare the assumptions made in the experimental studies with the assumptions made in our scalar IBM simulations and identify the physical systems that the current simulations represent.

### 3.2.1 Implications of simulation assumptions

All the unsteady models (Schumann, C-S, and D-C models) used in the experimental studies make the following assumptions: (1) neglect of radiative heat exchange between particles; (2) neglect of free convection effects; (3) same surface temperature for all the particles. The first two assumptions can be justified in certain parameter ranges based on simple scaling arguments that are presented below. The assumption that the particles equilibrate to the same surface temperature is based on the proximity of particles in packed beds. Only the more advanced experimental techniques that are able to measure surface temperature of each individual particle can prove or disprove this assumption.

The restrictions posed by the neglect of free convection and radiation heat transfer are now discussed in detail. Free convection is quantified by the Grashof number which is defined as

$$\text{Gr} = \frac{g\beta(T_f - T_s)D^3}{\nu_f^2} \quad (3.19)$$

where  $T_f$  is the free stream temperature,  $T_s$  is the temperature of the solid surface, and  $\beta$  ( $= 1/T_f$  for gases) is the volumetric thermal expansion coefficient. Free convection effects can be neglected if  $\text{Gr}/\text{Re}^2 < 1$  which implies

$$\hat{D} < \frac{\text{Re}^{2/3}}{(1 - T_s/T_f)^{1/3}}, \quad (3.20)$$

where  $\hat{D} = Dg^{1/3}/\nu_f^{2/3}$  is a non-dimensional diameter. This constraint imposes an upper limit on the particle diameter for each Reynolds number where free convection can be neglected. If an extreme value of 10 is taken for the fluid to solid temperatures ratio (i.e.,  $T_f/T_s = 10$ ), and air is assumed to be the fluid under terrestrial conditions ( $g = 9.81 \text{ m/s}^2$ ), then for the lowest  $\text{Re} = 0.001$  considered in this study the particle diameter has to be less than  $3.5 \mu\text{m}$  for negligible free convection. This restriction on the particle diameter becomes less severe as the Reynolds number increases. For example, for the highest Reynolds number of 300 considered in the accompanying paper (Garg et al., 2009a), free convection effects are negligible for particle diameter smaller than  $16 \text{ mm}$ . Since the analog of free convection is absent in mass transfer, the above limit (Eq. 3.20) on the regime of validity of scalar IBM simulations due to the neglect of free convection effects does not hold in the context of mass transfer.

For an isolated particle at  $T_s$  having emissivity equal to one, and surrounded by fluid at  $T_f$ , the ratio of radiation to forced convection heat transfer can be expressed as

$$q_{\text{rc}} = \frac{\sigma(T_s + T_f)(T_s^2 + T_f^2)}{h_{\text{fs}}} = \frac{\sigma D(T_s + T_f)(T_s^2 + T_f^2)}{\text{Nu} k_f}, \quad (3.21)$$

where  $\sigma = 5.67 \times 10^{-8} \text{ W/m}^2 \cdot \text{K}^4$  is the Stefan-Boltzmann constant. Assuming Air to be the surrounding fluid at  $T_s = 1000\text{K}$  ( $k_f = 0.060 \text{ W/m.K}$ ) and the particle is held at  $300\text{K}$ , and further assuming Stokes flow (i.e.,  $\text{Nu} \approx 2$ ), then the ratio of radiation to forced convection heat transfer  $q_{\text{rc}}$  is equal to 0.66 for millimeter sized particles. For particles smaller than millimeter, the ratio becomes even lesser (for example,  $q_{\text{rc}} = 6.60 \times 10^{-4}$  for micron sized particles). This

analysis has been performed in the Stokes flow regime. However, with increasing Reynolds number, the higher value of average Nusselt number further reduces the ratio of radiation to forced convection heat transfer; thus lessening the restriction on particle diameter. Therefore, from the analysis of both free convection and radiation heat transfer it is observed that restriction on particle diameter is most severe in the Stokes flow regime and eases with increasing Reynolds number.

Consistent with earlier work, we also assume that all the particles are maintained at the same temperature. This assumption, especially for dilute volume fractions, might not hold true in real flows. We also do not assume any conduction inside the particles, i.e., we assume infinite thermal conductance for particles. This is not an unreasonable assumption for many air–solid systems where the thermal conductivity of the particle is greater than that of air by more than an order of magnitude, e.g., air–coal, air–bismuth, air–gold, etc.

### 3.3 Solution Approach

The details of the hydrodynamic IBM solver have already been discussed in Chapter 2. In this section, the discussion is limited to implementation of the scalar solver for statistically homogeneous suspensions using the IBM. Similar to the approach taken for mass and momentum conservation equations, in IBM the scalar field is also solved in the entire domain that includes the interior regions of the solid particles as well. The scalar transport equation solved in IBM is

$$\rho_f C_{pf} \left\{ \frac{\partial \phi}{\partial t} + C_\phi \right\} = - \frac{\partial q_j}{\partial x_j} + S_{\phi,IBM} I_f + f_\phi, \quad (3.22)$$

where  $S_{\phi,IBM}$  is the sink/source term,  $C_\phi = \nabla \cdot \mathbf{u}\phi$  is the convective term in conservative form, and  $f_\phi$  is the additional immersed boundary force term that accounts for the presence of solid particles in the fluid–phase by ensuring desired isothermal boundary condition (Eq. 3.13) on the surface of the solid particles. Since the sink/source term needs to balance the interphase heat transfer between the two–phases (Eq. 3.15), it is applied only in the fluid–phase as a uniform volumetric term (i.e., the sink/source term does not depend on  $\mathbf{x}$ ).

The IB forcing  $f_\phi$  is computed only at points lying inside the solid particles which is

similar to the computation of  $\mathbf{f}_u$  for the hydrodynamic case. This ensures that, similar to the hydrodynamic forcing, the fluid–phase scalar field is not contaminated by the IB forcing. At the forcing points inside the solid, the fluid scalar field is forced in a manner similar to the ghost–cell approach used in standard finite–difference/finite–volume based methods. In addition to forcing the scalar field, the IB forcing term also cancels the remaining terms in the scalar transport equation and, at the  $n + 1^{\text{th}}$  time–step, it is calculated as

$$f_\phi^{n+1} = \rho_f C_{\text{pf}} \frac{\phi^d - \phi^n}{\Delta t} - \rho_f C_{\text{pf}} C_\phi^n + \frac{\partial q_j^n}{\partial x_j}. \quad (3.23)$$

where  $\phi^d$  is the desired scalar value at that location. Since the sink/source term  $S_{\phi, \text{IBM}}$  acts only in the fluid–phase, it does not appear in the above expression for IB forcing  $f_\phi$  which is computed only at points lying inside the solid particles.

For the problem of flow past a statistically homogeneous particle assembly, we solve the IBM governing equations by imposing periodic boundary conditions on fluctuating variables that are now defined. The schematic of this periodic computational domain in IBM simulations with multiple particles is shown in Fig. 2.2. The bounding surfaces of the computational domain, solid–phase, and fluid–phase are denoted by  $\partial\mathcal{V}$ ,  $\partial\mathcal{V}_s$ , and  $\partial\mathcal{V}_f$ , respectively. Due to periodic boundary conditions, it is possible that some particles intersect the boundaries of the computational domain. Therefore, the total bounding surface of the domain is decomposed as  $\partial\mathcal{V} = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}_f^{\text{ext}}$ , where  $\partial\mathcal{V}_s^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_s$  (shown by curly braces) is the domain surface cut by the solid particles, and  $\partial\mathcal{V}_f^{\text{ext}} = \partial\mathcal{V} \cap \partial\mathcal{V}_f$  (shown by dotted line) is the remaining domain surface. Therefore, the total bounding surface of the solid–phase is  $\partial\mathcal{V}_s = \partial\mathcal{V}_s^{\text{ext}} \cup \partial\mathcal{V}^{\text{int}}$ , where  $\partial\mathcal{V}^{\text{int}}$  (shown by solid lines) is along the actual surface area of the solid particles. Similarly, the total bounding surface of the fluid–phase is  $\partial\mathcal{V}_f = \partial\mathcal{V}_f^{\text{ext}} \cup \partial\mathcal{V}^{\text{int}}$ . The hatched lines represent the fluid–phase volume  $\mathcal{V}_f$  and the solid fill represents the solid–phase volume  $\mathcal{V}_s$ , such that the total volume of the computational domain  $\mathcal{V} = \mathcal{V}_s + \mathcal{V}_f$ .

From the definition of volumetric mean (Eq. 2.16), the scalar field can be decomposed as the sum of a volumetric mean and a fluctuating component

$$\phi(\mathbf{x}, t) = \langle \phi \rangle_{\mathcal{V}}(t) + \phi'(\mathbf{x}, t), \quad (3.24)$$

and similar decompositions can be written for the convective  $C_\phi$ , diffusive  $\mathbf{q}$ , and immersed boundary forcing  $f_\phi$  terms. The sink/source term is not decomposed because it is a spatially uniform volumetric term. Substituting the above decompositions in the scalar transport equation (Eq. 3.22), followed by volume averaging, yields the mean scalar transport equation, which is

$$\rho_f C_{\text{pf}} \frac{\partial \langle \phi \rangle_{\mathcal{V}}}{\partial t} = \varepsilon_f S_{\phi, \text{IBM}} + \langle f_\phi \rangle_{\mathcal{V}}, \quad (3.25)$$

where it is noted that due to periodic boundary conditions, the volume averages of the convective and diffusive terms are zero.

Subtracting the above mean scalar transport conservation equation (Eqs. 3.25) from the instantaneous scalar transport equation (Eqs. 3.22), yields the following conservation equation for the fluctuating scalar component:

$$\rho_f C_{\text{pf}} \left\{ \frac{\partial \phi'}{\partial t} + C'_\phi \right\} = - \frac{\partial q'_j}{\partial x_j} + f'_\phi(\mathbf{x}, t) + S_{\phi, \text{IBM}} I_f - \varepsilon_f S_{\phi, \text{IBM}}. \quad (3.26)$$

The conservation equations (Eqs. 3.23–3.26) are numerically solved to obtain the scalar field around immersed bodies that satisfies isothermal boundary condition (Eq. 3.13). The evolution of the mean scalar  $\langle \phi \rangle_{\mathcal{V}}$  given by Eq. 3.25 is a function of both the mean IB forcing  $\langle f_\phi \rangle_{\mathcal{V}}$  and sink/source term  $S_{\phi, \text{IBM}}$ . The mean IB forcing term  $\langle f_\phi \rangle_{\mathcal{V}}$  is computed by volume averaging the IB force computed by Eq. 3.23. The specification of sink/source term  $S_{\phi, \text{IBM}}$  is given in the next section.

### 3.3.1 Specification of the sink/source term $S_{\phi, \text{IBM}}$

In the previous section on governing equations (Sec. 3.2), it was shown that a volumetric sink/source term is necessary to obtain nontrivial steady-state solution to scalar transport in statistically homogeneous assemblies. The specification of sink/source term is analogous to specification of mean pressure gradient for the hydrodynamic case. In the hydrodynamic case (see Chap. 2 for details), the mean pressure gradient evolves in order to sustain the specified mean flow rate and prevents the fluid from coming to rest due to the friction posed by suspended particles. In a similar manner, the sink/source term evolves in order to maintain the desired

fluid-phase mean scalar value  $\langle \phi^{(f)} \rangle_{\mathcal{V}}^d$  and prevents the fluid scalar field from equilibrating to the surface scalar value of solid particles. The expression for true expectation of the sink/source term (Eq. 3.15) was derived earlier in Sec. 3.2 by ensemble averaging the scalar conservation equation 3.12 in the fluid-phase. The expression for sink/source  $S_{\phi,IBM}$  term in the IBM scalar conservation equation 3.22 can be derived by volume averaging this equation over the fluid-phase volume. However, during volume averaging over the fluid-phase volume one has to pay close attention to possible jump discontinuities. Following the steps take in previous chapter 2 for the fluid-phase volume averaging of the IBM momentum conservation equation 2.13, the fluid-phase volume average of the scalar transport equation 3.22 is

$$\rho_f C_{pf} \mathcal{V}_f \frac{d \langle u_i^{(f)} \rangle_{\mathcal{V}}}{dt} = \oint_{\partial \mathcal{V}^{int}} q_j^{(f)} n_j^{(s)} dA + S_{\phi,IBM} \mathcal{V}_f. \quad (3.27)$$

From Eq. 3.27, the sink/source term  $S_{\phi,IBM}$  that ensures the desired value for fluid-phase mean scalar  $\langle \phi^{(f)} \rangle_{\mathcal{V}}^d$  at the  $n^{\text{th}}$  time step is given by

$$S_{\phi,IBM}^n = \rho_f C_{pf} \frac{\langle \phi^{(f)} \rangle_{\mathcal{V}}^d - \langle \phi^{(f)} \rangle_{\mathcal{V}}^n}{\Delta t} - \frac{1}{(1 - \varepsilon_s) \mathcal{V}} \left\{ \oint_{\partial \mathcal{V}^{int}} \mathbf{q}^n \cdot \mathbf{n}^{(s)} dA \right\}, \quad (3.28)$$

where the superscript  $(f)$  in front of  $\mathbf{q}$  has been dropped for the sake of notation and it is assumed that scalar flux  $\mathbf{q}$  is evaluated on the fluid side of the fluid-particle interface, and the superscript  $n$  implies the relevant quantities at the time step. This equation for sink/source term at  $n^{\text{th}}$  time step is obtained by substituting a finite difference approximation for the unsteady term on left hand side of Eq. 3.27. This sink/source term  $S_{\phi,IBM}$  and the mean immersed boundary forcing term  $\langle f_{\phi} \rangle_{\mathcal{V}}$  (Eq. 3.23) are used to evolve the mean scalar by equation 3.25. For a statistically stationary flow, the equations are evolved in pseudo-time until the average quantities reach a steady state, at which point the first term on the right hand side of Eq. 3.28 is negligible, and Eq. 3.28 reduces to the numerical counterpart of Eq. 3.15. This establishes that the resulting numerical solution to the scalar IBM governing equations is a valid numerical solution to steady scalar transport past homogeneous particle assemblies.

### 3.4 Simulation Methodology

The setup of the hydrodynamic solver is the same as discussed in Chap. 2. For all the scalar transport simulations,  $\langle \phi^{(s)} \rangle_{\mathcal{V}}^d$  is set to zero by specifying zero scalar value at all particle surfaces. The desired fluid-phase mean scalar value  $\langle \phi^{(f)} \rangle_{\mathcal{V}}^d$  is equal to one.

In Sec. 2.4.1, the numerical parameters in scalar IBM simulations of homogeneous particle assemblies were identified as the box size to particle diameter ration  $L/D$ , grid resolution in terms of grid cells across a particle diameter  $D_m$ , number of particles  $N_s$ , and the time step  $\Delta t$ . Other than the choice of time step  $\Delta t$ , the other three numerical parameters do not change from the ones discussed in Sec. 2.4.1. In order to resolve the additional scalar diffusion time scale, the solution algorithm is advanced in pseudo-time from specified initial conditions to steady state using a time step  $\Delta t$  that is chosen as the minimum of the convective, viscous and diffusive time steps by the criteria

$$\Delta t = \text{CFL} \times \min \left\{ \frac{\Delta x}{u_{\max}}, \frac{\Delta x^2 (1 - \varepsilon_s)}{\nu_f}, \frac{\Delta x^2 (1 - \varepsilon_s)}{\alpha_f} \right\}, \quad (3.29)$$

where  $\alpha_f = k_f/\rho_f C_{pf}$  is the scalar diffusivity. At the beginning of the simulation  $u_{\max} = |\langle \mathbf{u}^{(f)} \rangle|$ , and as the flow evolves the time step adapts itself to satisfy the above criteria.

#### 3.4.1 Estimation of Nusselt number from simulations

DNS of flow through a particle using the IBM results in velocity, pressure, and scalar fields on a uniform Cartesian grid. Owing to deterministic particle position configuration in ordered arrays, one realization of the flow is adequate to ascertain drag force and Nusselt number. However, due to random particle position configurations in the random arrays, one has to perform multiple independent simulations in order to obtain faithful estimates of the same quantities. The average Nusselt number for the  $\mu^{\text{th}}$  realization is computed as

$$\{\text{Nu}\}_{\mathcal{V}}^{\mu} = - \frac{S_{\phi, \text{IBM}}^{\mu}}{k_f (\langle \phi^{(s)} \rangle_{\mathcal{V}} - \langle \phi^{(f)} \rangle_{\mathcal{V}})} \frac{1 - \varepsilon_s}{\varepsilon_s} \frac{D^2}{6}, \quad (3.30)$$

which is similar to the expectation of average Nusselt number (Eq. 3.17) derived earlier from averaged equations. The normalized average drag force  $F$  is similarly computed by integrating the viscous and pressure forces exerted by fluid on the particle surface (Chap. 2). The

statistical error resulting from using finite number of particles for random arrays is reduced by averaging this Nusselt number over  $\mathcal{M}$  multiple independent realizations that yields the ensemble-averaged “average Nusselt number”

$$\{\text{Nu}\}_{\nu, \mathcal{M}} = \frac{\sum_{\mu=1}^{\mathcal{M}} \{\text{Nu}\}_{\nu}^{\mu}}{\mathcal{M}}. \quad (3.31)$$

This ensemble-averaged “average Nusselt number” converges to the true expectation of the average Nusselt number  $\langle \text{Nu} \rangle$  (given by Eq. 3.18) in the limit  $N_s \mathcal{M} \rightarrow \infty$ .

In section 2.4.1 it is noted that each numerical parameter must be chosen to ensure numerically converged, accurate, and physically meaningful results. In the next section, we examine the numerical convergence of the IBM scalar solver with variation of numerical parameters. Similar to the steps taken in section 2.4.3 for the hydrodynamic solver, first the numerical convergence of Nusselt number with  $D_m$  is shown for ordered arrays. This is because for ordered arrays,  $L/D$  and the number of particles  $N_s$  are deterministic and the only remaining numerical parameter is the grid resolution  $D_m$ . Then attention is directed to random arrays where numerical convergence with respect all three parameters has to be considered in order to achieve a trade off between accuracy and computational effort.

### 3.5 Numerical convergence

Here we establish that scalar IBM simulations result in numerically converged solutions. The test case chosen is steady flow past an ordered array of particles in a lattice arrangement, because for this case the only numerical parameter is the grid resolution  $D_m$ . Although we consider steady flows, we also verify that the time step chosen to evolve the flow in pseudo-time from a uniform flow initial condition does not change the steady values of Nusselt number that we compute using IBM. For an FCC arrangement of particles, Fig. 3.2(a) shows the convergence characteristics of Nusselt number as a function of grid resolution  $D_m$  for two different solid volume fractions equal to 0.2 (triangles) and 0.4 (squares) at  $\text{Re} = 0.01$ . Fig. 3.2(b) shows the same comparison for FCC arrays at higher Reynolds number equal to 40. In both figures, open and filled symbols are for CFL values of 0.2 and 0.05, respectively. From both figures it

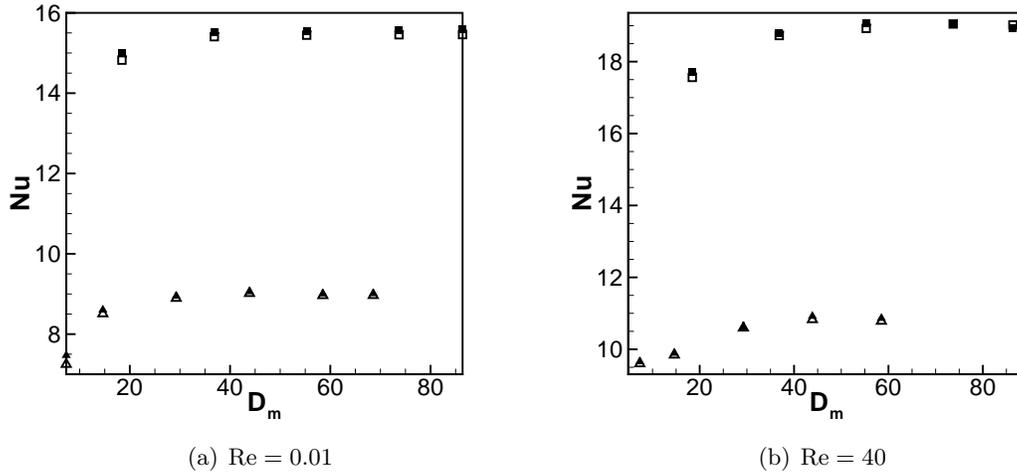


Figure 3.2 Convergence characteristics of Nusselt number with grid resolution  $D_m$  for FCC arrays for two different solid volume fractions equal to 0.2 (triangles) and 0.4 (squares) at (a)  $Re = 0.01$  and (b)  $Re = 40$ . Open and filled symbols are for CFL values of 0.2 and 0.05, respectively.

can be seen that the scalar IBM simulation result does not depend on the time step (CFL). With regard to spatial convergence, the figures show that the resolution requirements increase with increasing Reynolds number and volume fraction. For example, comparison of the spatial convergence of Nusselt number for  $\varepsilon_s = 0.4$  shows that while a minimum resolution of  $D_m = 20$  is needed for converged results at  $Re = 0.01$  (Fig. 3.2(a)), the minimum resolution requirement increases to  $D_m = 30$  at  $Re = 40$  (Fig. 3.2(a)). For the dependence of minimum resolution on solid volume fraction, Fig. 3.2(b) shows, for  $Re = 40$ , that the minimum resolution requirement increases from  $D_m = 20$  to  $D_m = 30$  as the solid volume fraction increases from 0.2 to 0.4, respectively. Due to the dependence of minimum numerical resolution on Reynolds number, for the scalar IBM simulations of ordered arrays that are reported later, grid resolutions higher than those shown in Fig. 3.5 are used for the same volume fractions so that the higher Reynolds number cases are also adequately resolved. It is noted here that the above resolution requirements have been established for a specific case of  $Pr = 0.7$ . As the Prandtl number increases, the thinning of thermal boundary layer with respect to the viscous boundary layer

will necessitate the use of higher numerical resolutions than those deemed sufficiently resolved for the current case.

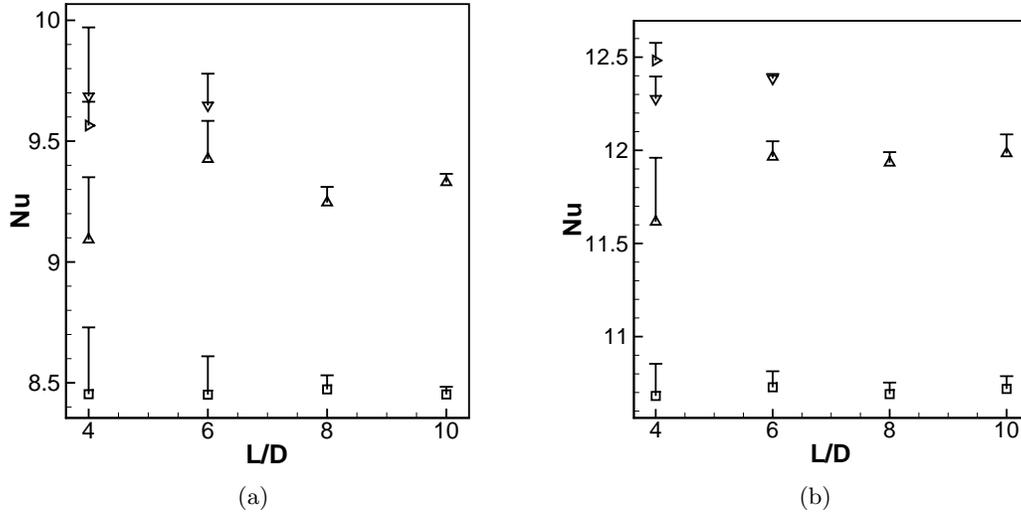


Figure 3.3 Convergence characteristics for random arrays at  $Re = 20$  of the Nusselt number with box length to particle diameter ratio  $L/D$  for four different values of  $D_m$  equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Solid volume fraction is equal to 0.3 in (a) and 0.4 in (b). Nusselt number values have been averaged over 5 multiple independent simulations. Not all combinations of  $D_m$  and  $L/D$  are shown because with a serial code some combinations exceeded computational memory requirements.

For the random arrays, in addition to errors arising from finite resolution, errors arise due to statistical fluctuations between different realizations and the box length is also an independent numerical parameter. Ideally, the effect of each numerical parameter on the numerical error should be investigated by varying that parameter while holding the other numerical parameters at fixed values. However, the choice of some numerical parameters must satisfy more than one requirement, and some error contributions are determined by the choice of more than one numerical parameter. Specifically, the choice of  $L/D$  is determined by more than one requirement (decay of spatial autocorrelation and the need for minimum number of samples in the average force estimate), and both  $L/D$  and the number of multiple independent simulations  $\mathcal{M}$  determine the number of samples in the force estimate. These considerations,

as well as computational limitations, did not permit the independent variation of numerical parameters. Therefore, a limited investigation of numerical parameter variation is presented here. To place this in context, we note that to our knowledge this is the most comprehensive study of numerical error and convergence for DNS of gas-solids flow.

While for ordered arrays the box length and number of particles are determined by the volume fraction and type of lattice arrangement (SC/FCC), in random arrays these parameters have to be carefully chosen. If  $L/D$  is too small, then the spatial autocorrelations that are larger than the box size will not be captured and the periodic images will interact. For steady flow past random arrays ( $\varepsilon_s = 0.3$ ,  $Re = 20$ ), Fig. 3.3(a) shows the convergence characteristics of the Nusselt number with box length to particle diameter ratio for four different values  $L/D$  equal to 10 (squares), 20 (upper triangles), 30 (lower triangles), and 40 (right triangles). Fig. 3.3(b) is the same comparison for a denser random array with volume fraction equal to 0.4. These results show that the Nusselt number value does not depend on  $L/D$ , and the effect of grid resolution  $D_m$  is stronger than  $L/D$  for the cases considered here. The simulations of flow past random arrays that are reported later in this work use higher resolutions when the Reynolds number exceeds 100%, as shown in Tab. 2.1.

In summary, these numerical convergence test results show that the scalar IBM simulations yield grid-independent results, and these results are also independent of the choice of time step used to advance the solution in pseudo-time, provided the stability criterion is met. The tests for random arrays also show that the grid-converged results do not exhibit a statistically significant dependence on the computational box length. However, these specific values for the numerical parameters should be treated as tentative because these limited set of tests cannot establish sharp limits on the minimum resolution required, and further numerical testing could refine these limits. A satisfactory number of MIS should ideally be determined by the determining the minimum number of samples for a given level of statistical error in the force estimate. However, this quantity is a strong function of  $Re$  and solid volume fraction. In the plots shown above, we have used 5 MIS for all the cases. While this results in a statistical error that is on the order of the other numerical error contributions, further testing is needed

to refine this requirement. Clearly, the requirements of minimum  $L/D$ , minimum  $D_m$ , and minimum  $\mathcal{M}$ , together dictate a trade-off for a fixed level of computational work. Of these parameters, our tests reveal that the numerical error in IBM exhibits the highest sensitivity to grid resolution  $D_m$ . These numerical convergence tests provide useful guidelines in the choice of these parameters that approximately balance the error contributions, but further testing is needed for a complete error analysis.

### 3.5.1 Numerical method validation

The hydrodynamic IBM solver was extensively validated in the last chapter by comparing drag force from IBM simulations for three different cases: (1) comparison of drag force for flow past single particle in an unbounded medium with the single sphere drag correlation of Schiller and Naumann (1933), (2) comparison of drag force in Stokes flow regime for flow past SC and FCC arrangements (ranging from dilute volume fraction to close packed limit) with the boundary–integral method of Zick and Homsy (1982), and (3) comparison of drag force for moderate to high Reynolds ( $Re \leq 300$ ) in SC and FCC arrangements with the LBM simulations of Hill et al. (2001a). For all these cases, the drag force from IBM simulations was found to be in excellent agreement with the existing literature. The lack of agreement in experimental literature about the average Stokes Nusselt number values and the lack of numerical studies preclude an extensive validation of the scalar solver.

In the light of the above facts, the flow over an isolated sphere in an unbounded medium presents itself as the logical validation test for any direct numerical simulation approach to gas-solid flow. However, especially for simulations that use periodic boundary conditions, this turns out to be a difficult validation test. For simulations using periodic boundary conditions, flow through a very dilute simple cubic arrangement is taken as a close approximation to flow over an isolated sphere in an unbounded medium. Since the simple cubic lattice arrangement is not isotropic, it is known (Hill et al., 2001a) that the results for drag can depend on the orientation of the flow with respect to the unit vectors of the lattice for values of Reynolds number beyond the Stokes flow regime. In contrast, there is of course no preferred direction for

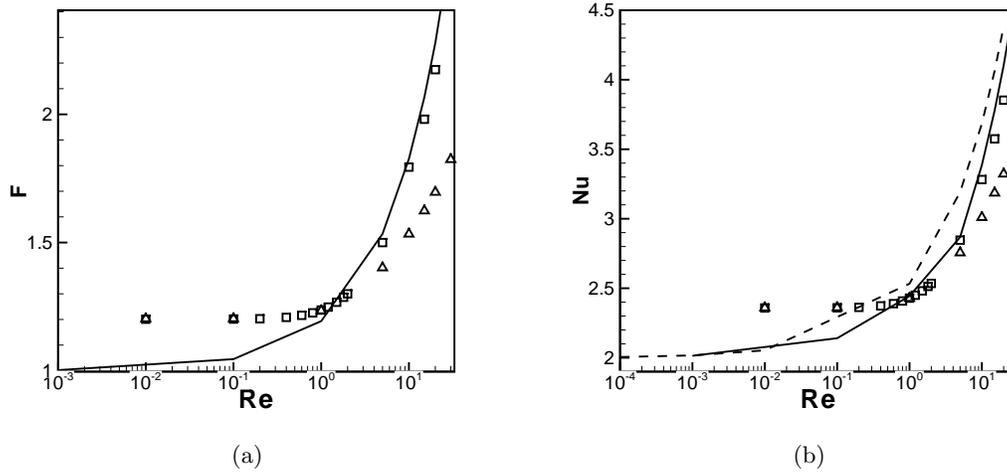


Figure 3.4 (a) Normalized drag force  $F$  and (b) Nusselt number in a simple cubic array ( $\varepsilon_s = 4.0E - 04$ ) as a function of Reynolds number and angle  $\theta$  between the mean flow and the  $x$ - axis in the  $(x, y)$  plane. The symbols are from the IBM simulations:  $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ ). Lines are a simple fit to the data. In (a) and (b), the solids lines are the single-sphere correlations for normalized drag force (Schiller and Naumann (1933)) and Nusselt number (Eq. 3.32), respectively. Also in (b), the dashed line is the Nusselt number from Gunn's correlation at infinite dilution ( $\varepsilon_s = 0$ ).

flow over an isolated sphere in an unbounded medium. For this simple problem, the normalized drag force is well approximated by the Schiller and Naumann (1933) correlation. For the heat transfer, the following Nusselt number correlation (Clift et al., 1978) is used to compare IBM results

$$\text{Nu} = \begin{cases} 2.0 + 0.6\text{Re}^{1/2}\text{Pr}^{1/3} & \text{if } \text{Re} < 1 \text{ (Ranz and Marshall, 1952),} \\ 1 + [1 + (1/\text{RePr})]^{1/3} \text{Re}^{0.41}\text{Pr}^{1/3} & \text{if } 1 \leq \text{Re} \leq 400. \end{cases} \quad (3.32)$$

Since Gunn's correlation (cf. Eq. 3.10) is valid at infinite dilution, it is also compared with the scalar IBM results.

Fig. 3.4(a) shows the comparison of normalized drag force  $F$  in a simple cubic array as a function of the Reynolds number from IBM simulations (hollow symbols) with the Schiller and Naumann drag correlation (solid line). Similarly, Fig. 3.4(b) shows the comparison of

average Nusselt number  $Nu$  from scalar IBM simulations (hollow symbols) with the correlation for average Nusselt number (solid line) for isolated particle (Eq. 3.32). In Fig. 3.4(b), the comparison is also made with average Nusselt number from Gunn's correlation (dashed line) at infinite dilution ( $\varepsilon_s = 0$ ). The drag and average Nusselt number computed for mean flow oriented at two different angles ( $\theta = 0$  ( $\square$ ),  $\theta = \pi/16$  ( $\triangle$ )) with respect to the lattice unit vector is shown to illustrate the dependence on flow angle. The sphere resolution ( $D_m$ ) for the simulation shown is equal to 12.8 grid cells. Fig. 3.4(a) is the same as Fig. 2.5 shown in the last chapter (Sec.2.5.1) and it has been reproduced here for the sake of comparison.

Comparison of Figs. 3.4(a) and 3.4(b) reveal that both the average Nusselt number and normalized drag force exhibit the same behavior with the mean flow angle and Reynolds number and two main conclusions can be drawn: (1) for  $Re > 1$ , the average Nusselt number and normalized drag force from IBM simulations are in good agreement with the existing average Nusselt number and drag laws for isolated particle only when the mean flow is directed at an angle of  $\pi/16$  in the  $(x, y)$  plane; (2) for  $Re < 1$ , the normalized drag force and average Nusselt number values are independent of the mean flow angle because the momentum and scalar transport are diffusion dominated, and diffusion is insensitive to the mean flow angle.

This simple test case is used to validate IBM's extension to scalar transport. We now move to the scalar IBM results for average Stokes Nusselt number  $Nu_0$ . The average Stokes Nusselt number is obtained for different solid volume fractions for ordered and random arrays, and compared with the existing heat and mass transfer correlation of Gunn (1978).

## 3.6 Results

### 3.6.1 Stokes flow

For the random arrays, the numerical resolutions given in Tab. 2.1 are used in scalar IBM simulations. For the ordered arrays, a constant grid size of  $150^3$  (i.e.,  $M = 150$  in Eq. 2.34) is used. Since the number of particles are known for ordered arrays (1 in SC, 4 in FCC), box length to particle diameter  $L/D$  ratio is deterministic for a given solid volume fraction  $\varepsilon_s$  from Eq. 2.33. Given the grid size  $M$  and box length to particle diameter ratio for a given solid

volume fraction, the grid resolution  $D_m$  can be computed from Eq. 2.34.

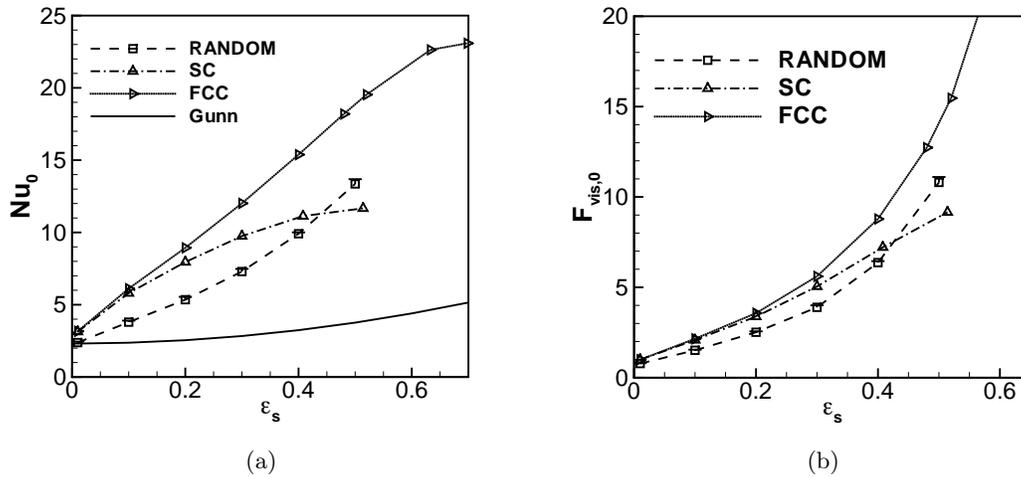


Figure 3.5 Comparison of (a) the average Stokes Nusselt number  $Nu_0$  and (b) the scaled (by Stokes drag) Stokes viscous drag force  $F_{vis,0}$  versus the solid volume fraction  $\epsilon_s$  between SC, FCC, and random arrangements in the Stokes flow regime.

Comparison of figures 3.5(a) and 3.5(b) shows two similarities: 1) the FCC arrangement results in the highest drag force and average Stokes Nusselt number for the entire range of solid volume fraction, 2) random arrays give lower average Stokes Nusselt number and drag force than the SC arrays for solid volume fraction up to approximately 0.41, beyond which they overtake the SC arrays for both the Nusselt number and viscous drag force.

For the average Stokes Nusselt number dependence on volume fraction, Fig. 3.5(a) shows that both FCC and SC arrays asymptote as the volume fraction approaches the respective close-packing limits (0.521 for SC and 0.744 for FCC). Although the maximum volume fraction simulated for random arrays is equal to 0.5 and far from the close-packing limit of 0.64, the dependence of average Stokes Nusselt number on  $\epsilon_s$  up to  $\epsilon_s = 0.5$  does not suggest an asymptotic limit as observed for ordered arrays. In section (3.6.4), the effect of particle-particle interactions on the average Stokes Nusselt number in ordered arrays is investigated by examining the local Nusselt number along the surface of the particles. In addition to the local Nusselt number, the local viscous drag along the surface of the particles is also examined. It

is shown that as the volume fraction approaches the close-packing limit in ordered arrays, the relative (relative to lower solid volume fractions) increase in local Nusselt number at the local maxima is compensated by a corresponding decrease at the local minima; thus, resulting in an asymptote for the average Nusselt number. For the Stokes viscous drag  $F_{\text{vis},0}$ , however, the local viscous drag increases monotonically with increasing solid volume fraction. Therefore, the Stokes viscous drag (Fig. 3.5(b)), unlike the average Stokes Nusselt number, does not asymptote in the limit of close-packing.

For the random arrays, apart from the number density and solid volume fraction, another important parameter in the particle configuration is the hard-core distance  $h_c$ , which is the minimum distance between the centers of any two particles. For the ordered arrays, for a given solid volume fraction, the hard-core distance is fixed and cannot be varied independently. However, for fixed volume fraction and number density it is possible to have random configurations that can correspond to different hard-core distances. From the study of local Nusselt number for ordered arrays in Sec. 3.6.4, it is shown that the local Nusselt number attains its maxima and minima in the regions of maximum and minimum flow channel widths. This motivates an investigation of the dependence of average Stokes Nusselt number  $\text{Nu}_0$  on the hard-core distance in random arrays. In Sec. 3.6.5, the dependence of average Stokes Nusselt number on hard-core distance is investigated.

Although not shown in this chapter but discussed in detail in the next chapter, the average Nusselt number from scalar IBM simulations remains nearly constant in the Stokes flow regime ( $\text{Re} < 1$ ). However, Gunn's correlation depends strongly on Reynolds number even in the Stokes flow regime, and decreases monotonically with decreasing Reynolds number. Therefore, in Fig. 3.5(a) the average Stokes Nusselt number from Gunn's correlation has been computed at  $\text{Re} = 0.01$ . Comparison of the average Stokes Nusselt number for random arrays from scalar IBM simulations with Gunn's correlation in Fig. 3.5(a) reveals a maximum difference of about 300% at the highest simulated solid volume fraction of 0.5. While proposing his correlation, Gunn (1978) argued through a stochastic model that inclusion of axial diffusion does not alter the asymptotic value of average Stokes Nusselt number. However, the earlier

frequency response measurements of Gunn and Desouza (1974) pointed to the importance of axial diffusion at low Reynolds numbers ( $\text{Re} < 10$ ). In order to verify the importance of axial diffusion in Stokes flow regime, in the next section we consider the budget of heat transport equation (Eq. 3.22) in fluid-phase along the mean flow direction, and show that the axial diffusion is one-third of the total diffusion.

### 3.6.2 Budget of the scalar transport equation

The diffusion term  $\nabla^2\phi$  in the scalar transport equation 3.22 can be decomposed as the sum of axial  $\frac{\partial^2\phi}{\partial\eta^2}$  and perpendicular diffusion  $\frac{\partial^2\phi}{\partial\xi^2}$ , where  $\boldsymbol{\eta} = \frac{\mathbf{u}}{|\mathbf{u}|}$  is a unit vector along the mean flow direction, and  $\boldsymbol{\xi}$  is a unit vector perpendicular to  $\boldsymbol{\eta}$  such that  $\boldsymbol{\eta} \cdot \boldsymbol{\xi} = 0$ , and  $\mathbf{q} = -k_f \nabla\phi$  has been substituted for scalar flux term in the scalar transport equation 3.22. Using this decomposition for the diffusion term, the steady-state heat transport equation in the fluid-phase becomes

$$\widehat{C}_\phi = \widehat{D}_{\phi,\parallel} + \widehat{D}_{\phi,\perp} + 1, \quad (3.33)$$

where  $\widehat{C}_\phi = \rho_f C_{\text{pf}} \frac{\nabla \cdot (\mathbf{u}\phi)}{S_\phi}$  is the normalized convection term, and  $\widehat{D}_{\phi,\parallel} = \frac{k_f}{S_\phi} \frac{\partial^2\phi}{\partial\eta^2}$  and  $\widehat{D}_{\phi,\perp} = \frac{k_f}{S_\phi} \frac{\partial^2\phi}{\partial\xi^2}$  are the normalized axial and perpendicular diffusion terms. Since this balance equation has been written for the fluid-phase only, the additional immersed boundary force term  $f_\phi$  in the original heat transport equation 3.22 drops out.

Since we are interested in evaluating the relative importance of the three terms in the above equation, the local fluid-phase volumetric average of each term can be defined along  $x$  in the  $y - z$  plane for the purpose of comparison. For ordered arrays, owing to the deterministic particle configurations, one realization will be sufficient. However, for random arrays, each term is a random process at any fixed location. Therefore, for the random arrays, the local fluid-phase volumetric average of the normalized axial diffusion term on the  $\mu^{\text{th}}$  realization  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}^\mu$  in the  $yz$  plane at any location  $x$  is defined as

$$\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}^\mu(x) = \frac{\int_{\mathcal{L}_y} \int_{\mathcal{L}_z} \widehat{D}_{\phi,\parallel}^\mu(x, y, z) I_f^\mu(x, y, z) d\mathcal{V}}{\int_{\mathcal{L}_y} \int_{\mathcal{L}_z} I_f^\mu(x, y, z) d\mathcal{V}}, \quad (3.34)$$

where  $d\mathcal{V} = dx dy dz$  is the infinitesimal volume at  $(x, y, z)$ , and  $I_f^\mu(x, y, z)$  is the fluid-phase indicator function at  $(x, y, z)$  on the  $\mu^{\text{th}}$  realization. Local fluid-phase volumetric averages for the normalized convection  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu(x)$  and perpendicular diffusion  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}^\mu(x)$  terms can be similarly defined. For random arrays, the local volumetric averages on each realization  $\left( \langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu, \langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}^\mu, \text{ and } \langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}^\mu \right)$  are further averaged over  $\mathcal{M}$  independent realizations to obtain the ‘‘ensemble averaged’’ local fluid-phase volumetric averages as  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}(x)$ ,  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}(x)$ , and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}(x)$ . For statistical homogeneous case the average statistics, strictly speaking, do not depend on position  $\mathbf{x}$ , and a global average over the fluid-phase volume will be sufficient. However, if the above quantities are further averaged along  $x$ -, then the average convective term  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu$  due to the use of periodic boundary conditions will be zero, i.e.,  $\int_{\mathcal{L}_x} \langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu(x) dx = 0$ . Since we are interested in finding out the relative importance of convection, axial and perpendicular diffusion terms in the scalar transport equation, the budget is studied along the mean flow direction  $x$ -, even for the statistically homogeneous case.

Fig. 3.6(a) shows the comparison of relative magnitudes of the ‘‘ensemble averaged’’ local fluid-phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}$  (dash-dot line) along the  $x$ - axis for  $\varepsilon_s = 0.1$  and  $\text{Re} = 0.01$ . Fig. 3.6(b) shows the same comparison for a denser solid volume fraction of  $\varepsilon_s = 0.4$  at  $\text{Re} = 0.01$ . Comparison of the figures reveal a negligible contribution from the convection term, which is expected in the Stokes flow regime. Gunn (1978) also neglected the convection term in the Stokes flow regime in his analysis.

If the ratio of axial diffusion to total diffusion is computed as  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}} / \left( \langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}} + \langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}} \right)$  in Figs. 3.6(a) and 3.6(b), then an average value of one-third is obtained for both cases. Although not shown here, the above observation (that axial diffusion is one-third of the total diffusion) holds true for all the solid volume fractions cases considered for both random and ordered arrays in Fig. 3.5(a). In his Stochastic model, Gunn (1978) had argued that the axial diffusion has no bearing on the average Stokes Nusselt number. However, the above analysis reveals that the axial diffusion is always one-third of the total diffusion for all values of  $\varepsilon_s$  in the Stokes flow regime. Therefore, the neglect of this term is not justified and, among vari-

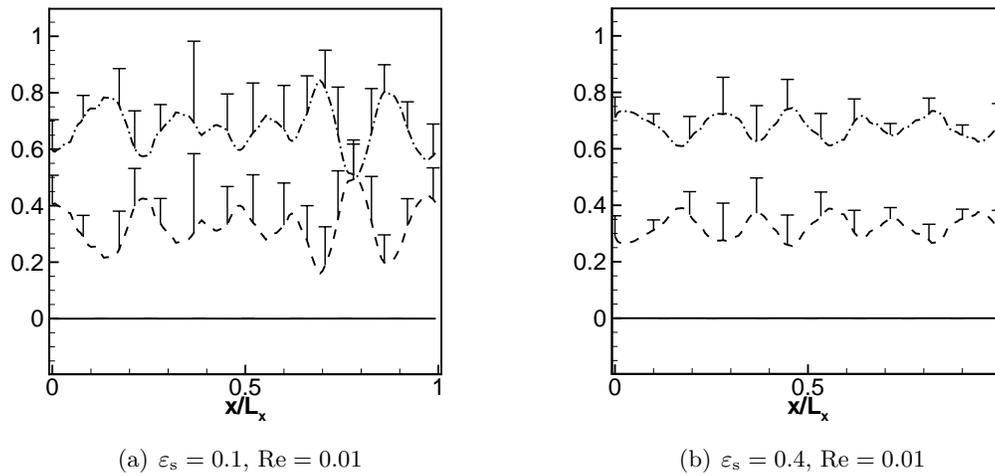


Figure 3.6 Comparison of relative magnitudes of the “ensemble averaged” local fluid-phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_\nu$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_\nu$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_\nu$  (dash-dot line) along the  $x$ -axis obtained for the normalized convection  $\widehat{C}_\phi$ , axial diffusion  $\widehat{D}_{\phi,\parallel}$  and perpendicular diffusion terms  $\widehat{D}_{\phi,\perp}$ , respectively, in the normalized fluid-phase scalar transport equation 3.33. Ensemble average has been performed over 5 independent realizations, and the scatter between the realizations is shown by the one-sided error bars.

ous other assumptions made in Gunn's analysis, it is one of the primary reasons for the large differences in average Stokes Nusselt number between the scalar IBM simulations and Gunn's correlation.

### 3.6.3 A correlation for average Stokes Nusselt number $Nu_0$

From the average Stokes Nusselt number  $Nu_0$  versus solid volume fraction  $\varepsilon_s$  data in Fig. 3.5, we can now propose a correlation for the average Stokes Nusselt number as a function of  $\varepsilon_s$  in random arrays. In the limit of infinite dilution,  $Nu_0$  should tend to the Nusselt number value of 2, which is the average Stokes Nusselt number for an isolated particle in unbounded medium. For the random arrays,  $Nu_0$  as a function of  $\varepsilon_s$ , shown in Fig. 3.5(a), can be approximated within less than 1% error by the following expression:

$$Nu_0(\varepsilon_s) = 2 + \frac{1}{(1 - \varepsilon_s)^3} \left( 10.35\varepsilon_s + 5.51\varepsilon_s^2 - 18.16\varepsilon_s^{3/2} + 1.63\varepsilon_s^{1/3} \right), \quad (3.35)$$

which satisfies the above constraint of  $Nu_0 = 2$  for  $\varepsilon_s = 0$ . Although the above correlation has been obtained for scalar IBM simulation performed up to  $\varepsilon_s = 0.5$ , based on the dependence of  $Nu_0$  on  $\varepsilon_s$  in Fig. 3.5, its extension up to the close-packing limit in random arrays will not be very erroneous.

It was earlier observed in Fig. 3.5(a) that the average Stokes Nusselt number in ordered arrays tends to an asymptote in the limit of close-packing. The Stokes viscous drag, however, does not show a similar trend and increases monotonically with increasing solid volume fraction. In the next section, we examine the local Nusselt number and viscous drag along the surface of the particles in ordered arrays in order to explain the different trends for average Nusselt number and Stokes viscous drag as the solid volume fraction approaches the close-packing limit.

### 3.6.4 Effects of inter-particle distance on the local Nusselt number and local viscous drag

Although for ordered arrays the hard-core distance is fixed for a given volume fraction, due to the deterministic particle configuration in ordered arrays, they serve as a useful tool to

visualize the effect of particle–particle interactions. Figs. 3.8(a) and 3.8(b) show the contour plots of non–dimensional scalar flux  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with the flow stream lines for SC arrays having solid volume fractions equal to 0.1 and 0.3, respectively. Figs. 3.8(c) and 3.8(d) show the same comparison for FCC arrays having solid volume fractions equal to 0.1 and 0.3, respectively. The contour plots are shown on a  $x$ - $y$  that lies half way along the  $z$ -axis, as shown by the schematic in Fig. 3.7.

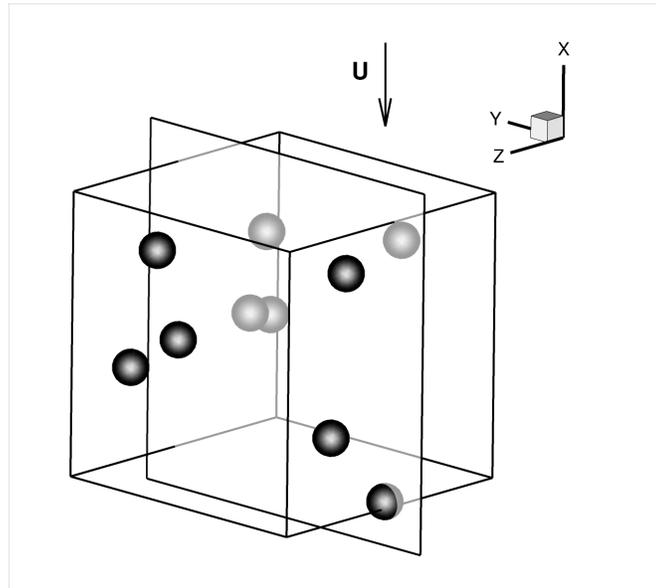


Figure 3.7 Schematic of the computational box along with solid particles. The contour plots of the non–dimensional scalar flux magnitude in the next few figures are shown in  $x$ - $y$  plane which lies midway along the  $z$ - axis.

For dilute volume fraction of 0.1, it can be seen from Figs. 3.8(a) and 3.8(c) that the flux is nearly constant along the particle surfaces. However, for a dense volume fraction of 0.3, Figs. 3.8(b) and 3.8(d) show that the flux peaks at points on the surface which are furthest from another particle surface. If a clockwise angle  $\phi$  is defined in the  $x$ - $y$  plane with respect to the negative  $x$ - axis, then for SC the peaks in the scalar flux occur at  $45^\circ + i90^\circ$ ,  $i = 0, 1, 2$ , and  $3$ . Figure 3.8(b) clearly shows that these are the points where the particle–particle distance is the highest. Similarly, for FCC arrays the peaks in the scalar flux occur

at  $i90^\circ$ ,  $i = 0, 1, 2$ , and  $3$ , which from Fig. 3.8(d) are the points of maximum particle–particle distance in FCC arrays. Comparison of the streamlines in the above figures reveals an interesting phenomena. Even in the Stokes flow regime, flow recirculation is observed in the SC arrangement (Fig. 3.8(b)). In contrast, for an isolated particle in an unbounded medium, flow recirculation is observed for  $Re > 20$ . However, the strong particle–particle interaction at finite volume fractions induces a recirculation even in the Stokes flow regime.

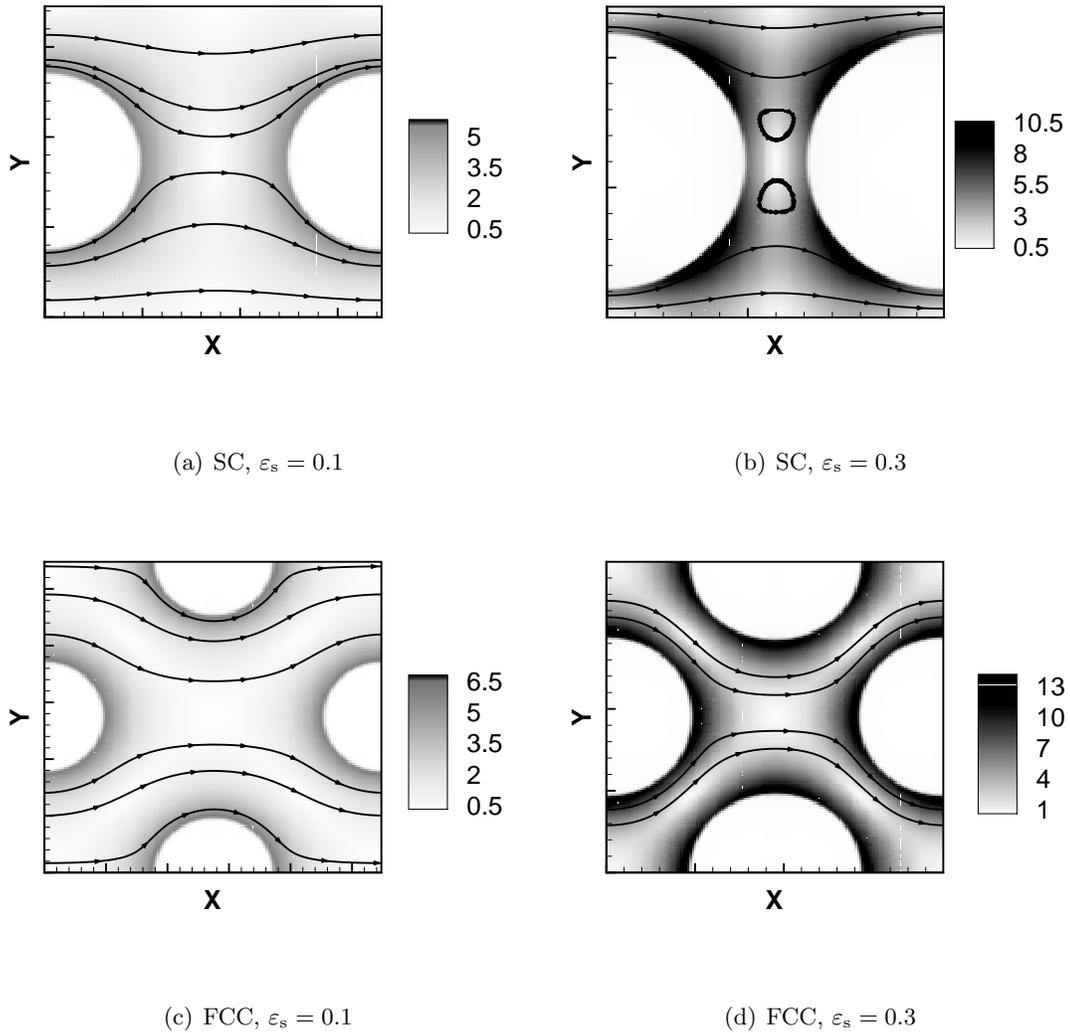


Figure 3.8 Contour plots of the non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for SC and FCC arrays at different volume fractions in the Stokes flow regime. The flow is directed along the  $x$ - (from left to right in the figures) axis and the contour plots are shown in the  $x$ - $y$  plane lying midway along the  $z$ - axis as shown by the schematic in Fig. 3.7.

This comparison of the contour plots for scalar flux shows that the inter-particle distance

can very strongly affect the local scalar flux along the particle surfaces. For Stokes flow (with isothermal boundary condition on surfaces), maximum fluxes (or scalar gradients) occur in regions with maximum fluid volume or wide flow channels. In regions with relatively less fluid volume between particle surfaces or narrow fluid channels, the scalar field equilibrates to the scalar surface value resulting in lower values for the scalar flux. In the presence of stronger convective transport (i.e., for higher Reynolds numbers), the resulting higher fluid velocity in narrow flow channels due to mass continuity will actually lead to higher fluxes (this is discussed in Sec. 4.1.1.1) in such regions than the wider flow channels. However, in the absence of strong convective transport in the diffusion dominated Stokes flow regime, the relatively higher equilibration of scalar fields in narrow fluid channels results in low local scalar fluxes along the particle surfaces bounding them.

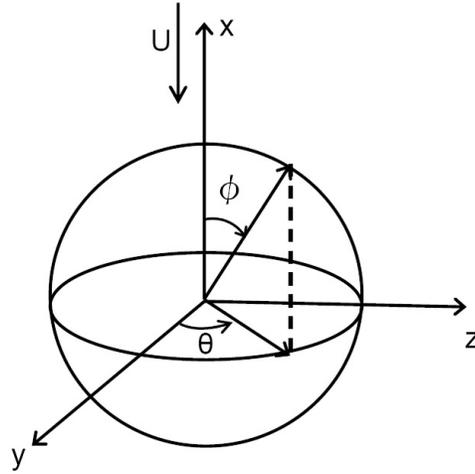


Figure 3.9 Schematic of the spherical coordinate system used to define local Nusselt number and local viscous drag.  $\phi$  ( $0 \leq \phi \leq \pi$ ) is the polar angle and  $\theta$  ( $0 \leq \theta \leq 2\pi$ ) is the azimuthal angle.

In the spherical coordinate system shown in Fig. 3.9, the local Nusselt number  $\text{Nu}^{\text{loc}}(\theta, \phi) = -\frac{\nabla \phi \cdot \mathbf{n}^{(s)}|_{r=R}}{\langle \phi^{(s)} \rangle - \langle \phi^{(f)} \rangle} D$  along the particle surface is a function of both the azimuthal angle  $\theta$  ( $0 < \theta < 2\pi$ ) and polar angle  $\phi$  ( $0 < \phi < \pi$ ). The local Nusselt number along the polar angle  $\phi$  is

obtained by integrating the local Nusselt number along  $\theta$  for each value of  $\phi$  as

$$\text{Nu}^{\text{loc}}(\phi) = \frac{\int_{\theta=0}^{2\pi} \text{Nu}^{\text{loc}}(\theta, \phi) d\theta}{2\pi}, \quad (3.36)$$

such that the average Nusselt number  $\text{Nu} = \frac{\int_{\phi=0}^{\pi} \text{Nu}^{\text{loc}}(\phi) d\phi}{\pi}$ . Similarly, the local Nusselt number along the azimuthal angle  $\theta$  is obtained by integrating the local Nusselt number along the polar angle  $\phi$  for each value of  $\theta$  as

$$\text{Nu}^{\text{loc}}(\theta) = \frac{\int_{\phi=0}^{\pi} \text{Nu}^{\text{loc}}(\theta, \phi) d\phi}{\pi}, \quad (3.37)$$

such that the average Nusselt number  $\text{Nu} = \frac{\int_{\theta=0}^{2\pi} \text{Nu}^{\text{loc}}(\theta) d\theta}{2\pi}$ . For ordered arrays, the local Nusselt number profile on each particle is identical, and therefore it suffices to show local Nusselt number along  $\theta$  or  $\phi$  for one particle only. However, for random arrays, all the particles in a realization are not identical. Therefore, in random arrays, the local Nusselt numbers are computed for each particle and then averaged over all the particles to yield a local average Nusselt number.

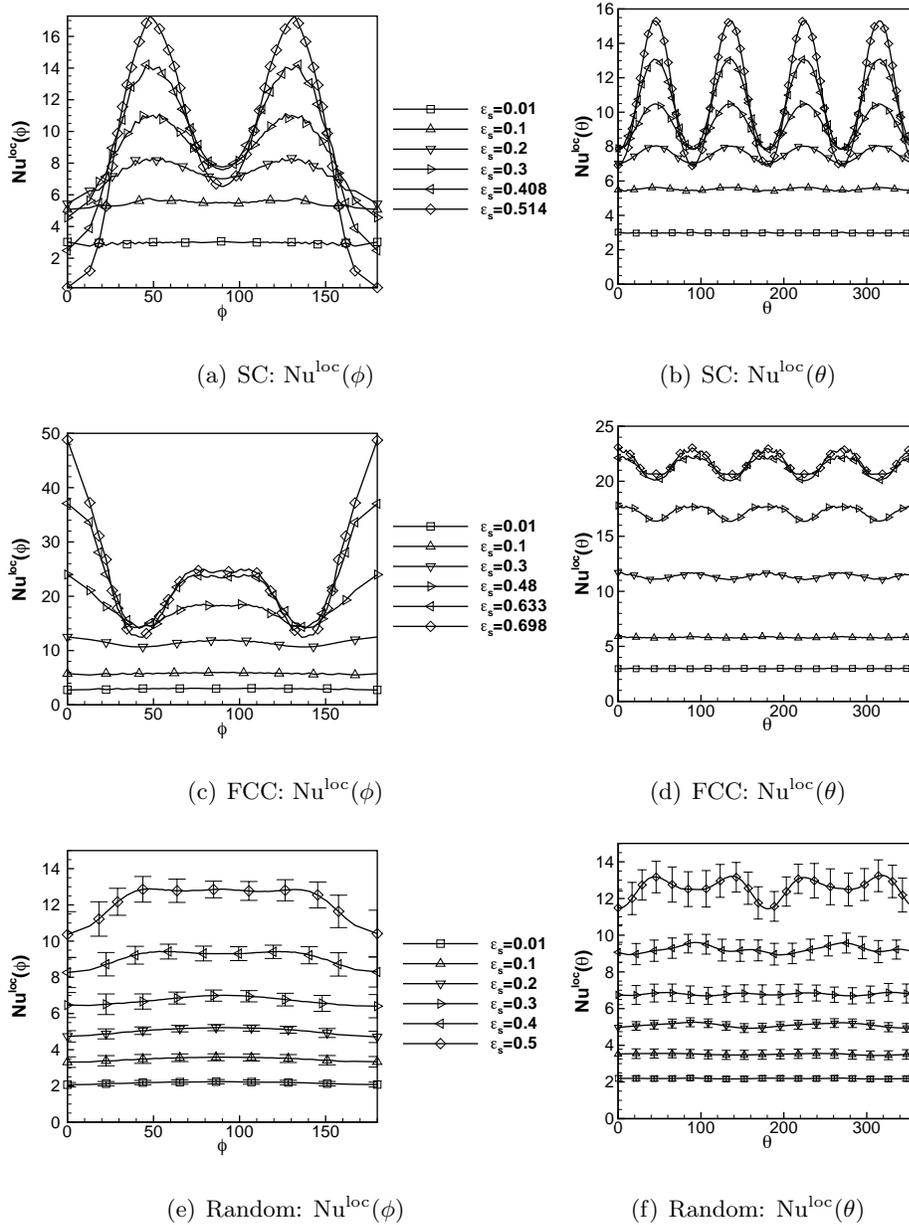


Figure 3.10 (a), (c), and (e): Comparison of the local Nusselt number  $Nu^{loc}(\phi)$  (Eq. 3.36) along the polar angle  $\phi$  for SC, FCC, and random arrays (local Nusselt number averaged over all bodies), respectively. (b), (d), and (f): Comparison of the local Nusselt number  $Nu^{loc}(\theta)$  (Eq. 3.37) along the azimuthal angle  $\theta$  for SC, FCC, and random arrays, respectively.

Figures. 3.10(a), 3.10(c), and 3.10(e) show the comparison of local Nusselt number  $Nu^{loc}(\phi)$  along the polar angle  $\phi$  for SC, FCC, and random arrays, respectively. For the random arrays, the local average Nusselt number obtained by averaging over all bodies is shown along with 95% confidence intervals. As observed earlier from the contour plots of scalar flux, the local Nusselt number does not depend on the polar angle  $\phi$  for dilute volume fractions. However, for dense volume fractions, the local Nusselt number peaks at  $45^\circ$  and  $135^\circ$  for the SC arrays, which correspond to the regions of maximum flow channel width in Figs. 3.8(a) and 3.8(b). For the FCC arrays, the local Nusselt number peaks at  $0^\circ$  and plateaus at  $90^\circ$  which also correspond to the regions of maximum flow channel width in Figs. 3.8(c) and 3.8(d). For the highest volume fractions of 0.514 and 0.698 considered for SC and FCC arrays, respectively, it can be seen that although the local Nusselt number increases relative to the lower volume fractions at the respective local maxima ( $45^\circ$  and  $135^\circ$  for SC,  $0^\circ$  and  $90^\circ$  for FCC), the local Nusselt number also decreases relative to the lower volume fractions at the local minima ( $0^\circ$  and  $90^\circ$  for SC,  $45^\circ$  and  $135^\circ$  for FCC). As a result, the Nusselt number  $Nu$  for SC and FCC arrays shows an asymptotic trend seen earlier in Fig. 3.5(a) as the solid volume fraction approaches the close-packed limit. However, for the random arrays the local average Nusselt number increases monotonically and the average Nusselt number  $Nu$  does not exhibit an asymptotic limit till the maximum solid volume fraction of 0.5 that is considered in this study.

Since the flow is directed along  $x$ - direction, the fore (at  $\phi = 0^\circ$ ) and aft (at  $\phi = 180^\circ$ ) symmetry of the flow is evident from the above plots of local Nusselt number  $Nu^{loc}(\phi)$  versus the polar angle  $\phi$ . The azimuthal angle  $\theta$ , however, varies in the plane perpendicular to the flow direction and, therefore, the local Nusselt number should be symmetric along  $\theta$ . Figures. 3.10(b), 3.10(d), and 3.10(f) show the comparison of local Nusselt number  $Nu^{loc}(\theta)$  (obtained from Eq. 3.37) along the azimuthal angle  $\theta$  for SC, FCC, and random arrays, respectively. Four main observations can be made from these figures: 1) the local Nusselt number  $Nu^{loc}(\theta)$  as expected is completely symmetric in all quadrants for SC and FCC arrays; 2) the local maxima of  $Nu^{loc}(\theta)$  occurs at  $45^\circ + i90^\circ$ ,  $i = 0, 1, 2$  and  $3$  for SC arrays and at  $i90^\circ$ ,  $i = 0, 1, 2$  and  $3$  for FCC arrays, which correspond to the regions of maximum flow chan-

nel width; 3) the increase in  $\text{Nu}^{\text{loc}}(\theta)$  at local maxima is offsetted by a decrease at the local minima, resulting in an asymptotic limit for average Nusselt number (Fig. 3.5(a)) as the solid volume fraction approaches the close-packed limit in SC and FCC arrays; 4) the local Nusselt number increases continuously for random arrays and, therefore, the average Nusselt number in Fig. 3.5(a) does not exhibit an asymptotic limit till the maximum solid volume fraction of 0.5 considered in this study. Comparison of the crests and troughs for the local Nusselt number for SC and FCC arrays reveals a relatively stronger polar and azimuthal variation in SC than in the FCC arrays.

Like the local Nusselt number  $\text{Nu}_{\text{vis}}^{\text{loc}}(\theta, \phi)$ , a local viscous drag can be defined as  $\mathbf{F}_{\text{vis}}^{\text{loc}}(\theta, \phi) = \mu_f (\nabla \mathbf{u} \cdot \mathbf{n}^{(s)}|_{r=R}) A$ , where  $A = \pi D^2$  is the solid particle surface area. Therefore, analogous to local Nusselt number, one can define a local viscous drag along polar angle  $\phi$  as

$$F_{\text{vis}}^{\text{loc}}(\phi) = \frac{\int_{\theta=0}^{2\pi} \mathbf{F}_{\text{vis}}^{\text{loc}}(\theta, \phi) \cdot \boldsymbol{\eta} d\theta}{2\pi F_{\text{Stokes}}}, \quad (3.38)$$

where  $F_{\text{Stokes}} = 3\pi\mu_f DU_{\text{slip}}(1 - \varepsilon_s)$  is the Stokes drag force and  $\boldsymbol{\eta}$  is the unit vector along flow direction (which is along  $x$ -direction for the current cases). The total viscous drag

$$F_{\text{vis}} = \frac{\int_{\phi=0}^{\pi} F_{\text{vis}}^{\text{loc}}(\phi) d\phi}{\pi}. \text{ Local viscous drag along the azimuthal angle } \theta \text{ can be similarly defined.}$$

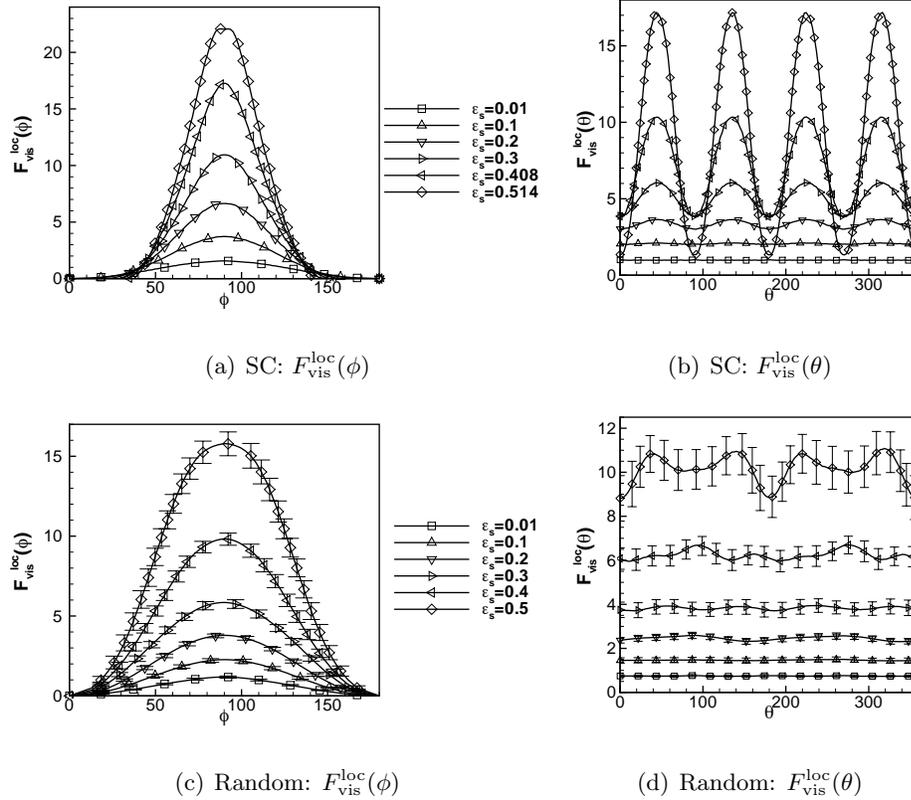


Figure 3.11 (a) and (c): Comparison of the local viscous drag  $F_{\text{vis}}^{\text{loc}}(\phi)$  (Eq. 3.38) along the polar angle  $\phi$  for SC and random arrays, respectively. (b) and (d): Comparison of the local viscous drag  $F_{\text{vis}}^{\text{loc}}(\theta)$  along the azimuthal angle  $\theta$  for SC and random arrays, respectively.

Figures. 3.11(a) and 3.11(c) show the comparison of local viscous drag  $F_{\text{vis}}^{\text{loc}}(\phi)$  along the polar angle  $\phi$  for SC and random arrays, respectively. It can be seen that the local viscous drag peaks at  $\phi = 90^\circ$  for all solid volume fractions for SC arrays. For the random arrays, just like in SC arrays, the local viscous drag (Fig. 3.11(c)) obtained by averaging over all particles in a realization peaks close to  $\phi = 90^\circ$ . However, unlike in SC arrays, the local viscous drag is finite over a wider range of polar angle in random arrays. This is because the local viscous drag on each particle in a single realization of random arrays exhibits significant scatter in terms of the maximum value of local viscous drag and its location. For example, Fig. 3.12 shows the

dependence of local viscous drag for 10 randomly chosen particles out of a total of 95 particles from a realization of random array with solid volume fraction  $\varepsilon_s = 0.4$ . It can be seen from the figure that there is a significant scatter both in terms of maximum local viscous drag and its location. The averaging over all the particles in a realization results in a bell shaped profile for local viscous drag in Fig. 3.11(c) for random arrays which is very similar to SC arrays in terms of location of the maximum but (due to particle scatter) is non-zero over a wider range of polar angle than in SC arrays.

In the local Nusselt number versus polar angle plots (Fig. 3.10), it was seen that as the volume fraction approached the close-packing limit in ordered arrays, the increase in local Nusselt number at the maxima was offsetted by the corresponding decrease at the minima; thus, resulting in an asymptote for the average Stokes Nusselt number  $Nu_0$  (Fig. 3.5(a)) in the limit of close-packing. However, in contrast, as seen in Fig. 3.11 the local viscous drag along the polar angle increases monotonically with solid volume fraction for both ordered and random arrays. As a result, the average Stokes viscous drag  $F_{vis,0}$ , unlike the average Stokes Nusselt number  $Nu_0$ , increases monotonically with solid volume fraction  $\varepsilon_s$  up to the close-packing limit in Fig. 3.5(b).

The peaking of viscous drag at  $90^\circ$  is reasonable and can be shown to hold true with an example of Stokes flow over an isolated sphere using the Oseen's (Oseen, 1910) approximation. If, as in the schematic shown in Fig. 3.9, the flow is directed along the negative  $x$ - axis, then the velocity field  $\mathbf{u}_{isol}^{Stokes}$  for Stokes flow over an isolated sphere from Oseen's approximation is

$$\mathbf{u}_{isol}^{Stokes} = -U \cos \phi \left[ 1 - \frac{3R}{2r} + \frac{R^3}{2r^3} \right] \hat{\mathbf{e}}_r + U \sin \phi \left[ 1 - \frac{3R}{4r} - \frac{R^3}{4r^3} \right] \hat{\mathbf{e}}_\phi, \quad (3.39)$$

where  $R = D/2$  is the radius of the particle,  $U$  is the free stream velocity, and  $\hat{\mathbf{e}}_r$  and  $\hat{\mathbf{e}}_\phi$  are the unit vectors along the radial and polar coordinates, respectively. The local viscous drag force  $F_{Stokes,vis}^{loc}$  on an isolated particle in Stokes flow along the polar angle  $\phi$  can be defined as

$$F_{vis,isol}^{loc}(\phi) = \mu_f (\nabla \mathbf{u}_{isol}^{Stokes} \cdot \mathbf{n}^{(s)}|_{r=R}) \cdot \boldsymbol{\eta} A. \quad (3.40)$$

In the spherical coordinate system, the surface normal vector is along the radial unit vector (i.e.,  $\mathbf{n}^{(s)} = \hat{\mathbf{e}}_r$ ), and  $\boldsymbol{\eta} = -\boldsymbol{\eta}_x = -\cos \phi \hat{\mathbf{e}}_r + \sin \phi \hat{\mathbf{e}}_\phi$ . From the definition of surface normal

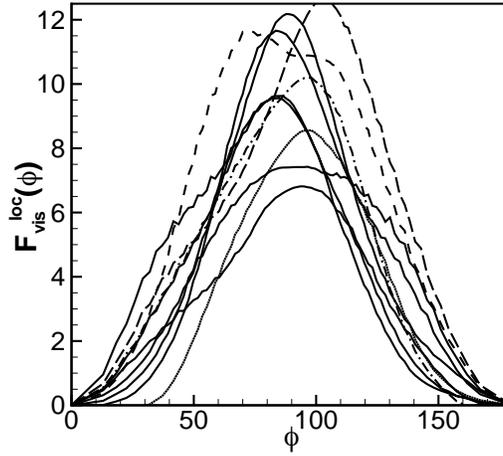


Figure 3.12 Comparison of the local viscous drag  $F_{\text{vis}}^{\text{loc}}(\phi)$  (Eq. 3.38) for 10 randomly chosen particles out of a total of 95 particles from a realization of random array with solid volume fraction  $\varepsilon_s = 0.4$ .

vector and velocity field (Eq. 3.39),  $\nabla \mathbf{u}_{\text{isol}}^{\text{Stokes}} \cdot \mathbf{n}^{(s)}|_{r=R} = \frac{3U}{2R} \sin \phi \hat{\mathbf{e}}_\phi$ . Therefore, the above local viscous drag force  $F_{\text{vis,isol}}^{\text{loc}}$  on an isolated particle becomes

$$F_{\text{vis,isol}}^{\text{loc}} = \left( \frac{3\mu_f U A}{2R} \sin \phi \hat{\mathbf{e}}_\phi \right) \cdot \hat{\mathbf{u}} = \left( \frac{3\mu_f U A}{2R} \sin \phi \hat{\mathbf{e}}_\phi \right) \cdot (-\cos \phi \hat{\mathbf{e}}_r + \sin \phi \hat{\mathbf{e}}_\phi) = \frac{3\mu_f U A}{2R} \sin^2 \phi. \quad (3.41)$$

The maxima of the above expression for  $F_{\text{vis,isol}}^{\text{loc}}$  lies at  $\phi = 90^\circ$ , and the minima lie at  $\phi = 0^\circ$  (fore) and  $\phi = 180^\circ$  (aft).

Figures. 3.11(b) and 3.11(d) show the comparison of local viscous drag  $F_{\text{vis}}^{\text{loc}}(\theta)$  along the azimuthal angle  $\theta$  for SC and random arrays, respectively. Since the azimuthal angle varies in the plane perpendicular to the flow direction, the local viscous drag profiles are symmetric in the four quadrants. For the SC arrays, similar to the observations for local Nusselt number  $\text{Nu}^{\text{loc}}(\theta)$ , the maxima of the local viscous drag lie at  $\theta = 45^\circ + i90^\circ$ ,  $i = 0, 1, 2$  and  $3$ , which correspond to the regions of maximum flow channel width (see Figs. 3.8(a) and 3.8(b) for SC arrays). For local viscous drag force  $F_{\text{vis}}^{\text{loc}}(\phi)$  versus polar angle  $\phi$ , very similar bell shaped profiles peaking at  $\phi = 90^\circ$  were observed for SC (Fig. 3.11(b)) and random arrays (Fig. 3.11(d)).

However, the local average viscous drag  $F_{\text{vis}}^{\text{loc}}(\theta)$  in random arrays, unlike in the SC arrays, is only weakly dependent on the azimuthal angle.

The above dependence of local Nusselt number on the flow channel width in ordered arrays motivates the study of dependence of average Stokes Nusselt number  $\text{Nu}_0$  on hard-core  $h_c$  distance in random arrays. Random arrays having same volume fraction and number density but with different hard-core distances can be generated using the Matèrn point-process or the three-step random configuration initialization algorithm outlined in Appendix A. Since with the three-step algorithm, higher values of maximum hard-core distance can be achieved than those possible by Matèrn point-process (about 1.5 times more, see Appendix A for details), the three-step algorithm is used here.

### 3.6.5 Effects of hard-core distance $h_c$ on average Stokes Nusselt number $Nu_0$ in random arrays

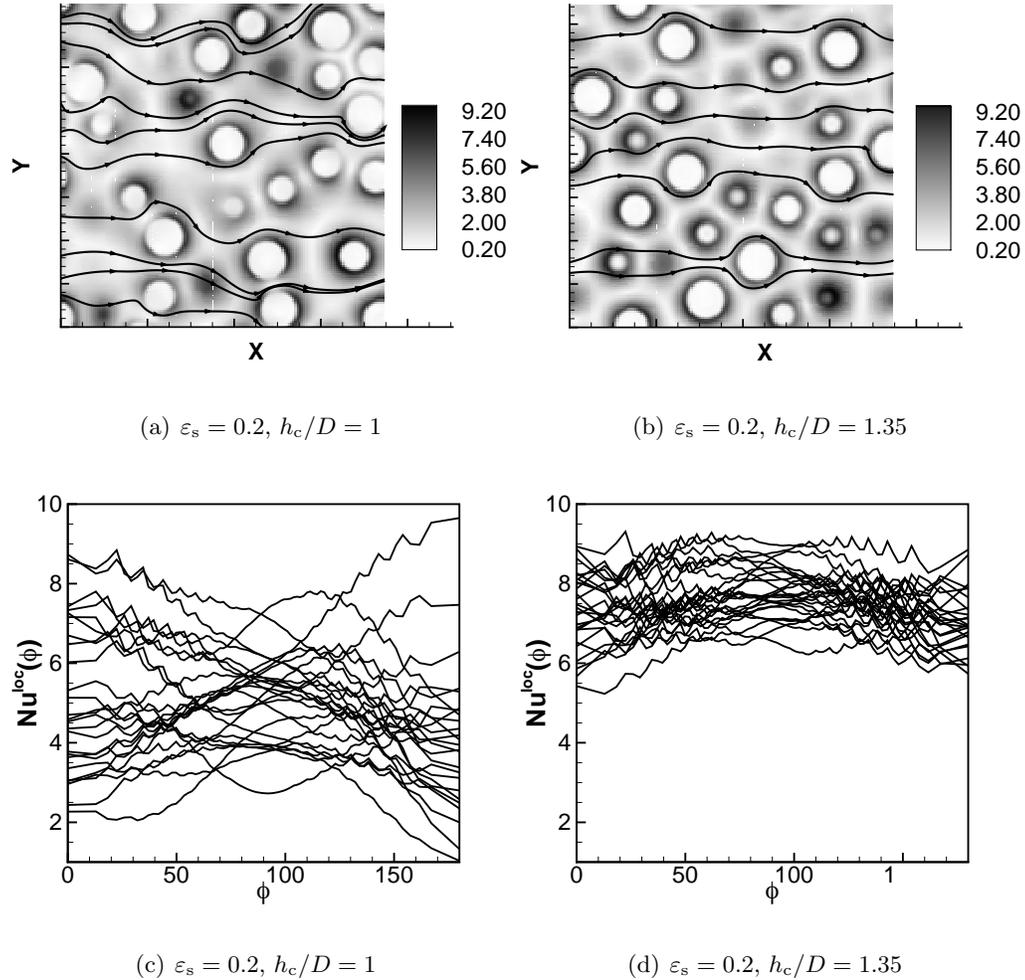


Figure 3.13 (a) and (b): Comparison of the contour plots of non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for random arrays with solid volume fraction  $\varepsilon_s = 0.2$  for hard-core distance  $h_c$  values equal to  $D$  and  $1.35D$ , respectively. (c) and (d): Comparison of the local Nusselt number  $Nu^{loc}(\phi)$  along the polar angle  $\phi$  on 20 randomly chosen particles for the same solid volume fraction and hard-core distances. Since the contour plots have been shown on a x-y plane lying midway along the z-axis (see schematic in Fig. 3.7), only the projected area of particles is visible in this plane.

Figs. 3.13(a) and 3.13(b) show the contour plots of non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  for random arrays (with solid volume fraction  $\varepsilon_s = 0.2$ ) for hard-core distance  $h_c$  equal to  $D$  and  $1.35D$ , respectively. Comparing the two figures, it can be seen that whenever two particles are in close proximity the scalar flux decreases at those surface points, i.e, the scalar flux reduces in the regions of narrow flow channels. This reduction in scalar flux is more severe for the case of  $h_c = D$  than  $h_c = 1.35D$  since the particles are allowed to be much closer (and hence narrower flow channels) in the former case. Due to the strong particle-particle interactions for  $h_c = D$ , and the resulting decrease of scalar flux in these regions, the local average Nusselt number ( $\text{Nu}^{\text{loc}}(\phi)$  from Eq. 3.36) for each body can drastically vary along  $\phi$ . Figs. 3.13(c) and 3.13(d) show the local average Nusselt number along polar angle  $\phi$  for 20 randomly chosen particles out of a total 161 particles from a realization of the random arrays ( $\varepsilon_s = 0.2$ ) for the same hard-core distances as above. Comparing the figures, it can be seen that the stronger particle-particle interactions for  $h_c = D$  result in strong dependence of local average Nusselt number on  $\phi$  for some particles. Furthermore, with increasing hard-core distance or weakening particle-particle interactions, the overall scatter in the local average Nusselt number also reduces. On an average, the local average Nusselt number is higher for particles for  $h_c = 1.35D$  than for  $h_c = D$ .

Figs. 3.14(a), 3.14(b), 3.14(c), and 3.14(d) show for  $\varepsilon_s = 0.01, 0.1, 0.2,$  and  $0.3$ , respectively, the local average Nusselt number  $\text{Nu}^{\text{loc}}(\phi)$  along  $\phi$  obtained by averaging over all bodies for different values of hard-core distance  $h_c$ . For each volume fraction, it can be seen that the local average Nusselt number increases with increasing hard-core distance or weakening particle-particle interactions. Also the error bars on local average Nusselt number shrink as the hard-core distance is increased. This is due to the reduction in scatter of local Nusselt number with increasing hard-core distance as seen and discussed above for  $\varepsilon_s = 0.2$  (Figs. 3.13(c) and 3.13(d)). By comparing the peaks of the local average Nusselt number, it can be seen that the effect of increased hard-core distance becomes stronger with increasing volume fraction. While for  $\varepsilon_s = 0.01$  the maximum local average Nusselt number increases by approximately 29% from 2.23 at  $h_c = D$  to 2.88 at  $h_c = 3.6D$ , for  $\varepsilon_s = 0.3$  it increases by approximately 41%

from 7.0 at  $h_c = D$  to 9.87 at  $h_c = 1.2D$ .

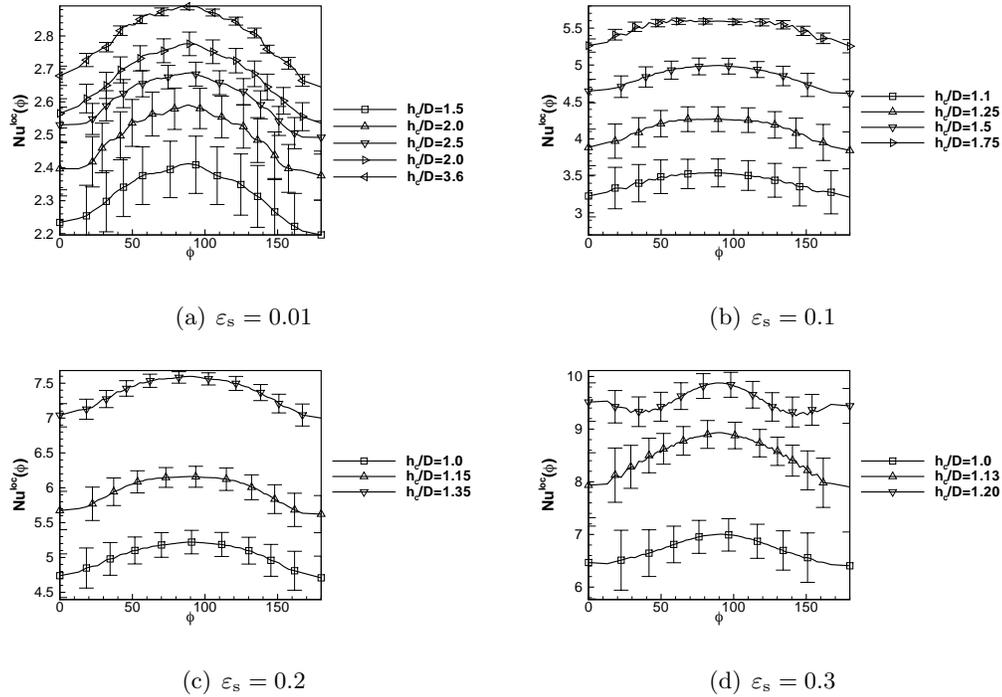


Figure 3.14 Comparison of the local average Nusselt number  $Nu^{loc}(\phi)$  (Eq. 3.36) along  $\phi$  obtained by averaging over all particles for different values of hard-core distance  $h_c$  in random arrays for (a)  $\varepsilon_s = 0.01$ , (b)  $\varepsilon_s = 0.1$ , (c)  $\varepsilon_s = 0.2$ , and (d)  $\varepsilon_s = 0.3$ .

Fig. 3.15(a) shows for  $\varepsilon_s = 0.01$  the comparison of average Stokes Nusselt number  $Nu_0$  as a function of hard-core distance by particle diameter ratio  $h_c/D$  for random arrays with the SC and FCC arrangements. It can be seen that the Stokes Nusselt number scales linearly (with a slope of 0.28) with the hard-core distance and increases by approximately 24% from  $h_c/D = 1$  to  $h_c/D = 3$ . As the hard-core distance is increased, Stokes Nusselt number for random arrays approaches the SC and FCC values. The same comparison at higher solid volume fractions of 0.1 (Fig. 3.15(b)), 0.2 (Fig. 3.15(c)), and 0.3 (Fig. 3.15(d)) once again reveal the linear scaling (with slopes equal to 2.8, 7.56, and 14.3, respectively) of Stokes Nusselt number with the hard-core distance. As observed earlier in local average Nusselt number versus the polar angle (Fig. 3.14), the effect of increasing hard-core distance becomes stronger with increasing solid

volume fraction which is evident from increasing slope magnitudes in Figs. 3.15(a)–3.15(d). For example, a ten fold increase in solid volume fraction from 0.01 to 0.1 results in a similar order of magnitude increase in slope from 0.28 to 2.8, respectively. The different pair-correlation functions  $g(r)$  corresponding to different hard-core distances for  $\varepsilon_s = 0.01, 0.1, 0.2,$  and  $0.3$  are shown in Figs. A.2(a), A.2(b), A.2(c), and A.2(d), respectively.

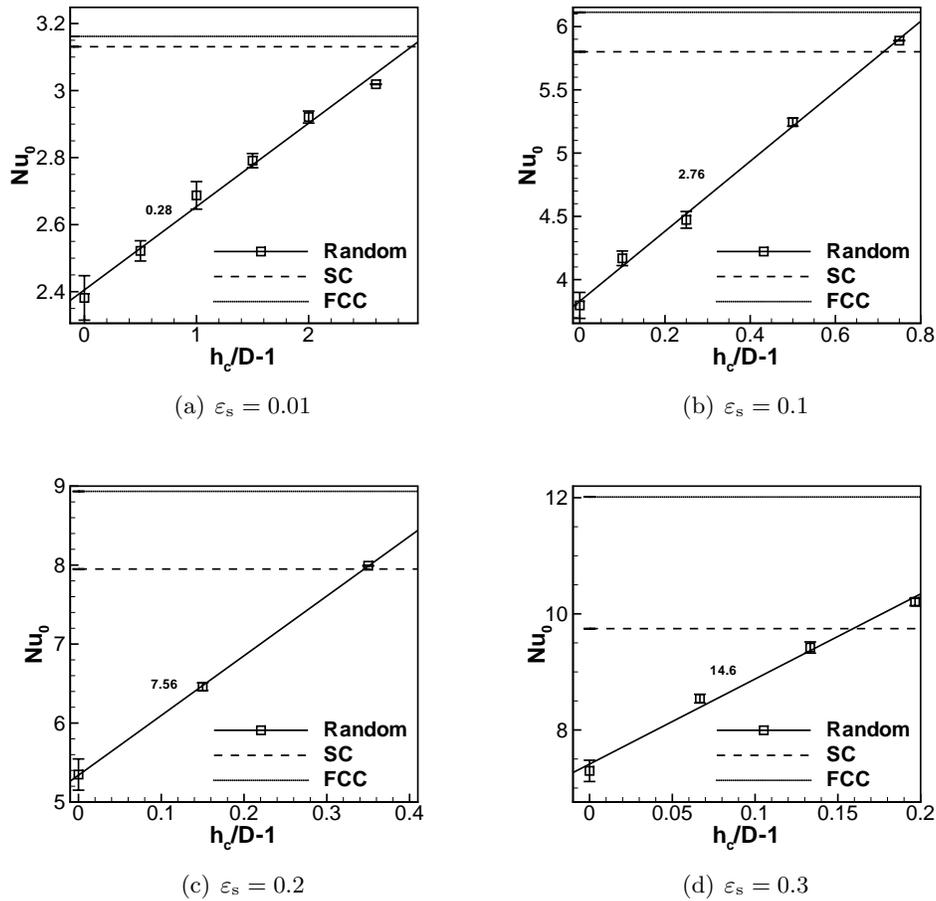


Figure 3.15 Comparison of the average Stokes Nusselt number versus hard-core distance  $h_c$  in random arrays with SC and FCC arrays for solid volume fraction  $\varepsilon_s$  equal to 0.01 in (a) and 0.1 in (b). The slopes of the solid lines obtained by linear least-squares fit is indicated next to them.

Based on the above observations, a modified average Stokes Nusselt number  $Nu'_0$  that, in addition to the first-order solid volume fraction effects, also accounts for the second-order

inter-particle distance effects can be decomposed as

$$\text{Nu}'_0(\varepsilon_s, \hat{h}_c) = \text{Nu}_0^{\hat{h}_c=1}(\varepsilon_s) + \alpha_{\text{Nu}}^{h_c}(\hat{h}_c - 1), \quad (3.42)$$

where  $\hat{h}_c = h_c/D$  is the ratio of hard-core distance and particle diameter,  $\text{Nu}_0^{\hat{h}_c=1}$  is the average Stokes Nusselt number when hard-core distance equals the particle diameter (i.e.,  $\hat{h}_c = 1$ ), and  $\alpha_{\text{Nu}}^{h_c}$  is the coefficient of the correction term to  $\text{Nu}_0^{\hat{h}_c=1}$  for finite hard-core distance. Since  $\text{Nu}_0^{\hat{h}_c=1}$  is the average Stokes Nusselt number for  $h_c = D$ , its functional dependence on solid volume fraction  $\varepsilon_s$  is already known from the earlier proposed correlation for  $\text{Nu}_0$  (Eq. 3.35). The coefficient  $\alpha_{\text{Nu}}^{h_c}$  in the above equation corresponds to the slopes of linear fits (solid lines) in Fig. 3.15. Based on the above discussion for Figs. 3.14 and 3.15, the hard-core distance correction reduces with decreasing solid volume fraction and logically it should vanish in the limit of infinite dilution, i.e.,  $\alpha_{\text{Nu}}^{h_c} \rightarrow 0$  as  $\varepsilon_s \rightarrow 0$ . Therefore, in the limit of infinite dilution, the above modified average Stokes Nusselt number  $\text{Nu}'_0$ , like  $\text{Nu}_0$  in Eq. 3.35, also tends to the isolated sphere limit of 2.

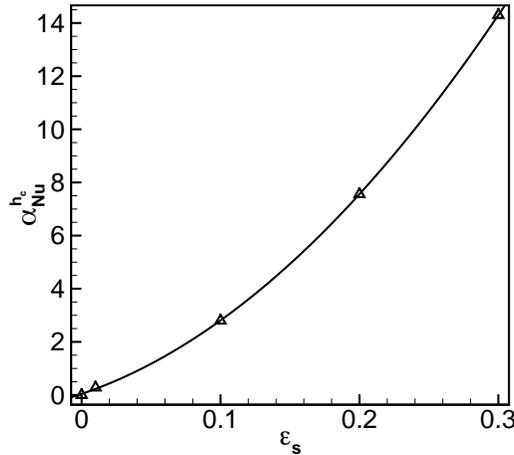


Figure 3.16 The correction term due to finite hard-core distance  $\alpha_{\text{Nu}}^{h_c}$  (Eq. 3.42) in the Stokes flow versus solid volume fraction  $\varepsilon_s$ . The solid line obtained as a quadratic fit is  $17.7\varepsilon_s + 99.39\varepsilon_s^2$ .

The hard-core distance  $h_c$  correction term  $\alpha_{\text{Nu}}^{h_c}$ , which corresponds to the slopes in Figs. 3.15(a)–

3.15(d), is shown as a function solid volume fraction  $\varepsilon_s$  in Fig. 3.16. In the figure, the hard-core distance correction term satisfies the earlier identified constraint that  $\alpha_{\text{Nu}}^{h_c} \rightarrow 0$  in the limit of infinite dilution. By fitting a quadratic polynomial to the data, the following functional dependence of  $\alpha_{\text{Nu}}^{h_c}$  on  $\varepsilon_s$  is obtained

$$\alpha_{\text{Nu}}^{h_c}(\varepsilon_s) = 17.77\varepsilon_s + 99.39\varepsilon_s^2, \quad (3.43)$$

which vanishes at  $\varepsilon_s = 0$ . With the above approximate fits for  $\text{Nu}_0^{\hat{h}_c=1}$  (Eq. 3.35) and  $\alpha_{\text{Nu}}^{h_c}$  (Eq. 3.43), the modified average Stokes Nusselt number  $\text{Nu}'_0$  (Eq. 3.42) is now completely known as a function of both the solid volume fraction  $\varepsilon_s$  and hard-core distance  $h_c$  by particle diameter  $D$  ratio for  $\text{Pr} = 0.7$ .

### 3.7 Discussion and Conclusions

A novel computational strategy based on direct-forcing immersed boundary method is devised to simulate heat/mass transfer in steady flow past homogeneous assemblies using periodic boundary conditions. DNS results are used to provide accurate closure for interphase heat/mass transfer term in averaged heat/mass conservation equations. The numerical convergence of scalar IBM solver in terms of grid resolution  $D_m$  and box length to particle diameter ratio  $L/D$  is established, and the solver is validated by comparing the average Nusselt number in a very dilute SC array ( $\varepsilon_s = 4.0E - 04$ ) with the average Nusselt number correlation for an isolated particle .

The dependence of the average Stokes Nusselt number  $\text{Nu}_0$  on solid volume fraction is compared for both the ordered (SC and FCC) and random arrays for a fixed Prandtl number of 0.7. For the random arrays, in addition to the solid volume fraction, dependence of average Stokes Nusselt number on the inter-particle spacing, quantified by the hard-core distance  $h_c$ , is also investigated.

For the ordered arrays, the average Stokes Nusselt number increases with solid fraction from a minimum value of 2.0 (corresponding to the isolated sphere limit) to a maximum value of  $\text{Nu}_{0,\text{max}}$  (approximately 11 for SC and 23 for FCC). As the volume fraction approaches

the close-packing limit, the average Stokes Nusselt number approaches an asymptotic limit in ordered arrays. This is because as the solid volume fraction increases, the increase in local Nusselt number at the local maxima is compensated by the corresponding decrease at the local minima.

While the ordered arrays show a sharper increase in average Stokes Nusselt number in the 0.01 – 0.4 solid volume fraction range and then level off, the random configuration for  $h_c = D$  shows a more gradual increase in the same range and no levelling off for the range of volume fractions considered. For the highest solid volume fraction of 0.5 simulated for random arrays, the average Stokes Nusselt number from scalar IBM simulations and Gunn's correlation differ by more than 300%. Through a comprehensive budget study of the convection and diffusion terms (which is further decomposed into axial and perpendicular diffusion terms) in the scalar transport equation, it is shown that axial diffusion is one-third of the total diffusion for all values of solid volume fractions. The assumption of negligible axial diffusion, among many other assumptions made in the stochastic model used by Gunn, is attributed to be the primary incorrect assumption that has resulted in differences as high as 300% for the average Stokes Nusselt number between the scalar IBM simulations and Gunn's correlation.

With the aid of ordered arrays, the effect of inter-particle distance on the local Nusselt is demonstrated. In ordered arrays, the local Nusselt number peaks in regions of maximum flow channel widths. Therefore as the average inter-particle distance is increased in random arrays by varying the hard-core distance  $h_c$ , the average Stokes Nusselt number increases. The dependence of average Stokes Nusselt number on  $h_c$  becomes stronger as the solid volume fraction increases and vanishes in the limit of infinite dilution. A modified average Stokes Nusselt number  $Nu'_0$  (Eq. 3.42) that accounts for the dependence of average Stokes Nusselt number on both the solid volume fraction  $\varepsilon_s$  and hard-core distance by particle diameter ratio  $\hat{h}_c$  is defined and a correlation proposed.

## CHAPTER 4. SCALAR TRANSPORT AND HEAT TRANSFER PAST ORDERED AND RANDOM ARRAYS OF MONODISPERSE SPHERES IN FLOWS WITH FINITE FLUID INERTIA

This chapter is a manuscript in preparation titled “Scalar transport and heat transfer past ordered and random arrays of monodisperse spheres for low to high Reynolds numbers” authored by R. Garg, S. Tenneti, and S. Subramaniam.

Direct numerical simulations of steady passive scalar transport in flow past stationary simple cubic (SC), face-centered cubic (FCC), and random arrays of monodisperse spheres are performed for Reynolds numbers from 1 to 300. The scalar DNS is performed using a computational approach that extends the immersed boundary method to solve for steady scalar transport past homogeneous particle assemblies. The average Nusselt number is compared with the widely-used heat and mass transfer correlation of Gunn (1978). It is found that for low solid volume fractions ( $\varepsilon_s < 0.1$ ), average Nusselt number from scalar IBM simulations and Gunn’s correlation agree well in random arrays. For higher solid volume fractions of random arrays, there is large difference between scalar IBM simulations and Gunn’s correlation, reaching as much as 300% for  $\varepsilon_s = 0.5$ . However, for random arrays the scalar IBM simulations and Gunn’s correlation converge nearly to the same values for higher solid volume fractions as the Reynolds number increases. This is attributed to the incorrect asymptote for average Stokes Nusselt number in Gunn’s correlation, which is addressed in the previous chapter 3 on average Stokes Nusselt number. Based on the scalar IBM simulations, a correlation for average Nusselt number as function of solid volume fraction and Reynolds number is proposed.

## 4.1 Results

Since the results for scalar transport/heat transfer in the Stokes flow regime show significant qualitative and quantitative differences (Chap. 3), we expect similar trends for the convection dominated finite Reynolds number regime. In the first subsection the average Nusselt number in ordered arrays is analyzed as a function of solid volume fraction and Reynolds number, and compared with Gunn's correlation. In order to explain the dependence of average Nusselt number on solid volume fraction and Reynolds number, we examine the local Nusselt number on the particle surface for selected cases. Contour plots of the scalar and velocity fields also provide insight into the underlying physical mechanisms of scalar transport/heat transfer. In the second subsection, the scalar transport/heat transfer in random arrays is examined. For the random arrays, a budget study of the scalar transport equation reveals that axial diffusion is always one-third of the total diffusion for all combinations of  $(\varepsilon_s, Re)$ .

### 4.1.1 Ordered arrays

#### 4.1.1.1 Simple cubic arrays

For SC arrays, Fig. 4.1(a) shows for SC arrays the comparison of average Nusselt number versus Reynolds number for different values of solid volume fraction  $\varepsilon_s$ . The comparison is also made with average Nusselt number from Gunn's correlation for random arrays for two extreme values of solid volume fraction equal to 0.01 (solid line) and 0.5 (dashed line) that encompass the solid volume fractions considered for SC arrays. Also shown in the same plot is the comparison with Nusselt number correlation for isolated particle (dash-dotted line, given by Eq. 3.32) from Clift et al. (1978). It can be seen that the maximum difference between IBM simulations (up to 300% for  $\varepsilon_s = 0.5$  and  $Re = 0.01$ ) and Gunn's correlation occurs in the  $Re$  range  $0 < Re < 10$ . While the IBM simulations predict an asymptote for average Nusselt number group in the Stokes flow limit ( $Re < 1$ ), the average Nusselt number from Gunn's correlation shows a significant dependence on  $Re$  in this range. For example, for  $\varepsilon_s = 0.5$ , while the average Nusselt number from Gunn's correlation increases by approximately 39% from 4.07 at  $Re = 0.01$  to 5.75 at  $Re = 1.0$ , IBM simulations predict a near constant value of

approximately 11.66 for the same range of Reynolds number.

From Fig. 4.1(a) it is observed that as the Reynolds number increases, differences between Gunn's correlation and scalar IBM simulations reduce and then increase again. This is due to the different slopes for average Nusselt number from scalar IBM simulations and Gunn's correlation. Since Gunn's correlation is primarily for random arrays, and the current comparison is made with ordered arrays, different slopes can be attributed to the configurational dependence. For the ordered arrays, the drag force for  $Re > 5$ , as shown in Hill et al. (2001b) and Garg et al. (2009c), is strongly dependent on the orientation of the mean flow. Since the scalar gets convected along the flow streamlines, the average Nusselt number is affected by orientation of the mean flow. This is shown by a limited comparison for two mean flow angles for very dilute SC array in scalar IBM validation in the previous chapter 3. Therefore, this comparison for ordered arrays is only limited due to the strong dependence of average Nusselt number on the mean flow angle for  $Re > 5$ .

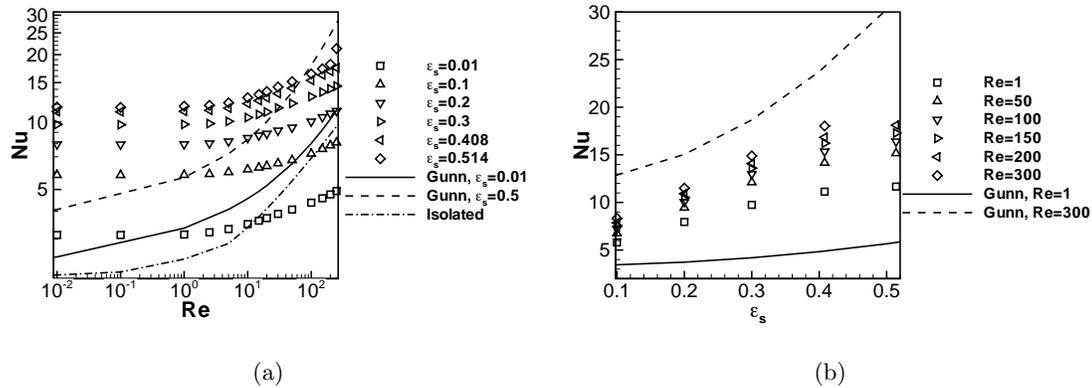


Figure 4.1 Comparison of the average Nusselt number in SC arrays (a) versus the Reynolds number  $Re$  for different values of solid volume fraction, and (b) versus the solid volume fraction  $\varepsilon_s$  for different values of Reynolds number with Gunn's correlation (solid and dashed lines in both (a) and (b)) and Nusselt number correlation (dash-dotted line) for isolated particle (Eq. 3.32) in (a). The flow is directed along  $x$ -axis for all the above cases.

Comparison with the isolated particle Nusselt number correlation (Eq. 3.32 from Clift

et al. (1978)) in Fig. 4.1(a) reveals that over a certain range of Reynolds number for  $\varepsilon_s = 0.01$  ( $Re > 15$ ) and 0.1 ( $Re > 200$ ), the average Nusselt number is less than that for an isolated particle. Similar observations are made for the drag force in Hill et al. (2001b) and in Fig. 2.7 (chapter 2) for  $F$  versus  $Re$  in SC arrays. In Fig. 2.7, it can be seen that the normalized drag force  $F$  over a range of Reynolds number for solid volume fractions 0.01 ( $Re > 10$ ) and 0.1 ( $Re > 100$ ) is less than that on an isolated particle. It is a well known fact that a particle shielded along the flow direction by an upstream particle experiences a lesser drag force. The drag force on a particle is the sum of forces resulting from the “short range” viscous and the “long range” pressure gradient forces. Due to the additional “long range” pressure gradient forces, the affect of shielding is more pronounced for the drag force than for the average Nusselt number which is clearly evident on comparing the slope of  $F$  versus  $Re$  in Fig. 2.7 against the slope of  $Nu$  number versus  $Re$  in Fig. 4.1(a).

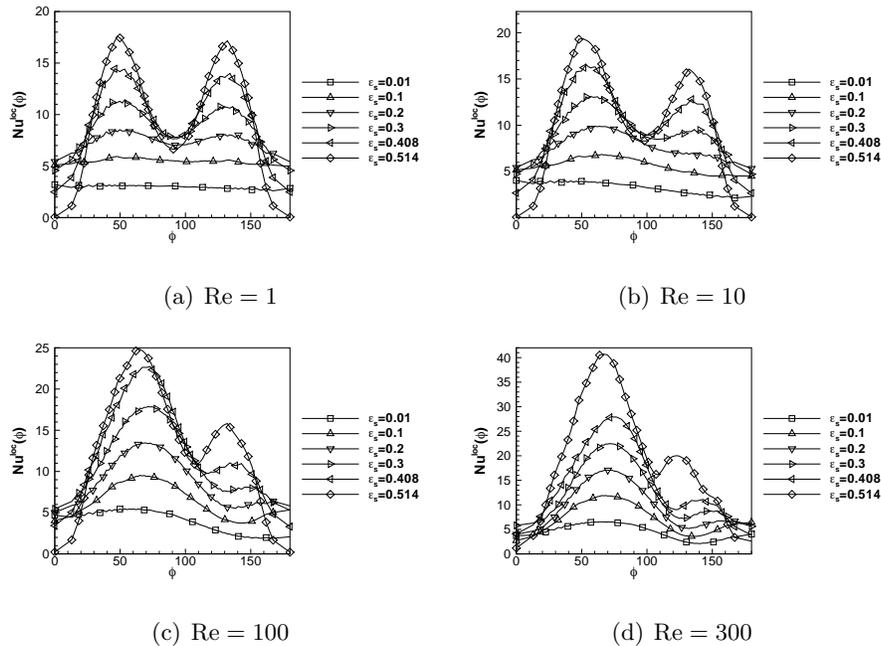


Figure 4.2 Comparison for different solid volume fractions in SC arrays of the local Nusselt number  $Nu^{loc}(\phi)$  (Eq. 3.36) along the polar angle  $\phi$  for  $Re$  equal to (a) 1, (b) 10, (c) 100, and (d) 300.

Figure 4.1(b) shows the comparison of average Nusselt number versus the solid volume

fraction  $\varepsilon_s$  for different Reynolds numbers. The comparison is also made with the average Nusselt number predicted by Gunn's correlation for two extreme values of Reynolds number equal to 1 (solid line) and 300 (dashed line) that encompass the Reynolds number range considered for SC arrays. It can be seen that for all values of Reynolds number the average Nusselt number from DNS approaches a asymptotic limit as the volume fraction approaches the close-packing limit. For Stokes flow at high volume fractions, it is observed in chapter 3 that the relative increase in local Nusselt number at the local maxima is offset by the relative decrease at the local minima, resulting in an asymptote for average Nusselt number in the limit of close-packing. For SC arrays, Figs. 4.2(a), 4.2(b), 4.2(c), and 4.2(d) show the local Nusselt number  $Nu^{loc}(\phi)$  (see Eq. 3.36 for the definition of  $Nu^{loc}(\phi)$ ) for different solid volume fractions as a function of the polar angle  $\phi$  for Re equal to 1, 10, 100, and 300, respectively. It can be seen that for all Reynolds numbers considered, the relative increase in local Nusselt number at the local maxima is compensated by the relative decrease at local minima. This results in an asymptotic limit for the average Nusselt for all Reynolds numbers in the limit of close-packing.

For  $Re = 1$  (Fig. 4.2(a)), the local Nusselt number dependence on  $\phi$  is similar to that observed for Stokes flow in chapter 3 with maxima occurring approximately at  $\phi = 45^\circ$  and  $135^\circ$  for  $\varepsilon_s \geq 0.2$  – which correspond to the regions of maximum flow channel width. As the Reynolds number increases to 10 (Fig. 4.2(b)), the maxima of the local Nusselt number still occur at approximately the same values of  $\phi$  equal  $45^\circ$  and  $135^\circ$ . However, unlike for Stokes flow and  $Re = 1$  cases, the second peak at  $\phi = 135^\circ$  is smaller in magnitude than the first peak at  $\phi = 45^\circ$ . Comparison at higher Reynolds numbers of 100 (Fig. 4.2(c)) and 300 (Fig. 4.2(d)) reveals an even greater drop in magnitude of the second peak relative to magnitude of the first peak. Comparison of the figures also show that the location of the first peak shifts from the regions of maximum flow channel width ( $\phi = 45^\circ$  and  $135^\circ$ ) in Stokes flow regime (Fig. 4.2(a)) toward the region of minimum flow channel width ( $\phi = 90^\circ$ ) as the Reynolds number increases.

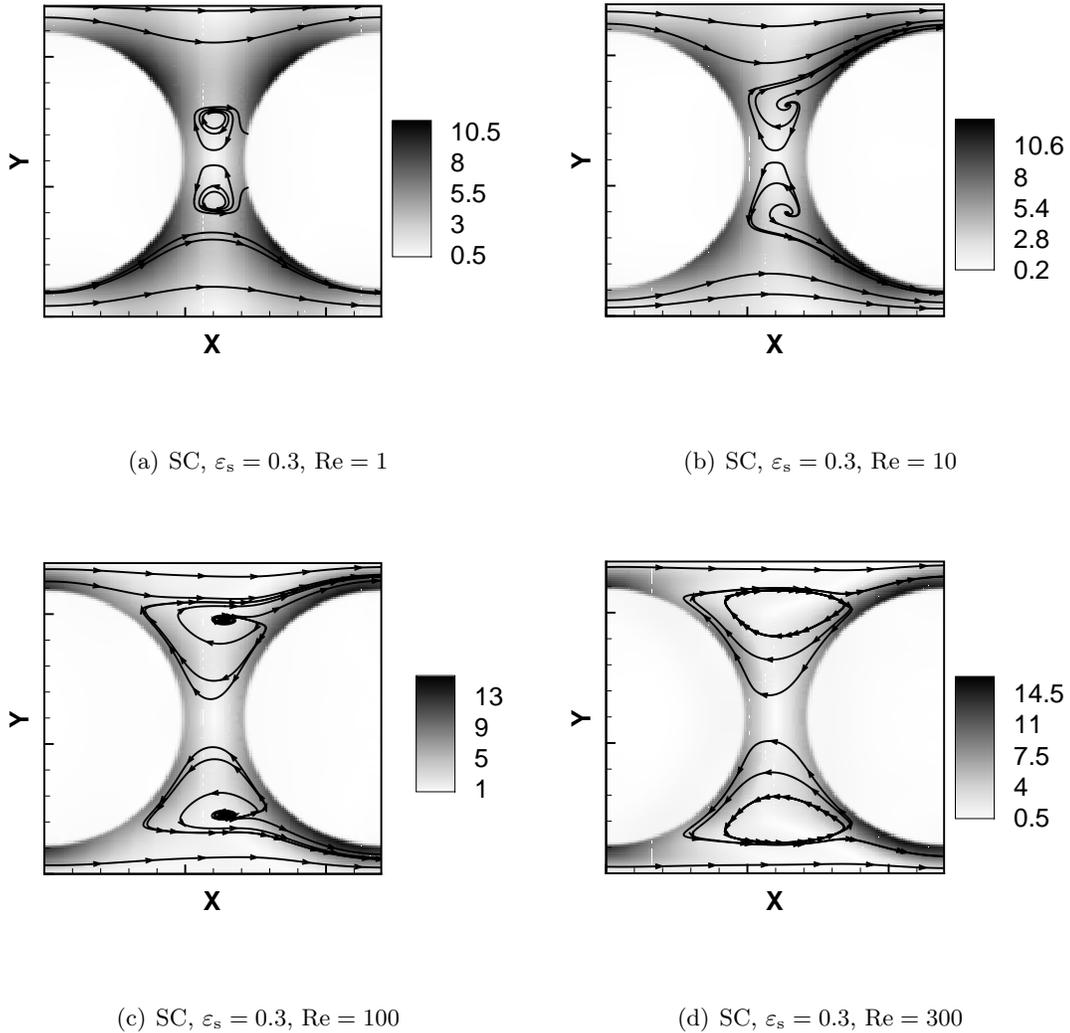


Figure 4.3 Contour plots of the non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with flow stream lines (shown by solid lines) for SC arrays with  $\varepsilon_s = 0.3$  at (a)  $Re = 1$ , (b)  $Re = 10$ , (c)  $Re = 100$ , and (d)  $Re = 300$ . The flow is directed along  $x$ - axis and the contour plots are shown in the  $x$ - $y$  plane lying midway along the  $z$ - axis as shown by the schematic in Fig. 3.7. Few streamlines in (a) seem to be penetrating the particle surface. This is an artifact of the visualization software that draws streamlines based on interpolated values from the Eulerian grid, resulting in finite but negligible fluid velocity even at fluid-particle interface.

The shifting of the first peak toward  $\phi = 90^\circ$  is due to the dominance of convective transport over diffusive transport as the Reynolds number increases. In the Stokes flow regime which is dominated by diffusive transport, maximum scalar gradients or heat transfer occur where there is maximum fluid volume between the particle surfaces. However, as the Reynolds number increases the high fluid velocity in thin regions between particles results in increased convective transport which eventually dominates over the diffusive transport as the Reynolds number increases.

The drop in magnitude of the second peak relative to the first peak with increasing Reynolds number is due to the widening of flow recirculation bubble at the rear as shown by the comparison of contour plots of non-dimensional scalar flux magnitude  $|\nabla\phi|D/|\langle\phi^{(s)}\rangle - \langle\phi^{(f)}\rangle|$  along with the flow streamlines for  $\varepsilon_s = 0.3$  at different values of Reynolds number in Fig. 4.3. Comparison of the streamlines shows that the span of recirculation bubble widens along  $\phi$  as the Reynolds number increases, thus causing lesser heat transfer along the portion of the particle surface exposed to the recirculation bubble.

Having studied at the average Nusselt number dependence on Reynolds number and solid volume fraction for SC arrays, we now move to the second type of ordered arrays considered in this work: FCC arrays.

#### 4.1.1.2 Face-centered cubic arrays

Figure 4.4 shows for FCC arrays the dependence of average Nusselt number  $Nu$  on Reynolds number for different solid volume fractions  $\varepsilon_s$ . Since Gunn's correlation is valid up to the close-packing limit in random arrays ( $\approx 0.65$ ), comparison of the scalar IBM simulations with his correlation is limited to a maximum solid volume fraction of 0.65. As observed earlier in the case SC arrays, the maximum difference between IBM simulations and Gunn's correlation occur in the range  $0 < Re < 10$ . For example, for  $\varepsilon_s = 0.633$  at  $Re = 10$ , Gunn's correlation and scalar IBM simulation differ by more than 100%.

Only a limited comparison can be made for the case of ordered arrays due to the additional dependence of average Nusselt number on the mean flow angle. It can be seen that, similar to

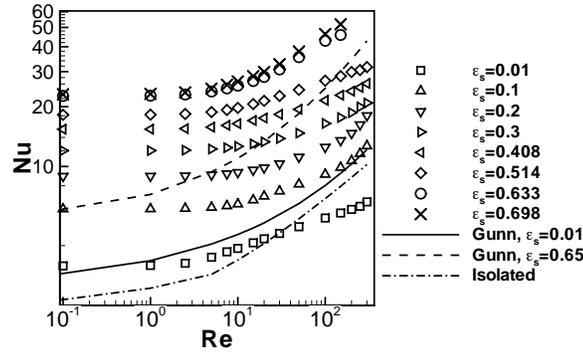


Figure 4.4 Comparison of the average Nusselt number in FCC arrays versus the Reynolds number  $Re$  for different values of solid volume fraction with Gunn's correlation (solid and dashed lines) and Nusselt number correlation (dash-dotted line) for isolated particle (Eq. 3.32). The flow is directed along  $x$ -axis for all the above cases.

observations made for SC arrays, the difference between IBM simulations and Gunn's correlation first reduces with increasing Reynolds number and then increases again. This is due to the different slopes for average Nusselt number for FCC arrays from scalar IBM simulations and Gunn's correlation. Similar to SC arrays, the average Nusselt number in FCC arrays also remains nearly constant in the Stokes flow regime, i.e.  $Re \leq 1$ . Gunn's correlation, on the other hand, is strongly dependent on  $Re$  in the Stokes flow regime and decreases monotonically. Also, similar to SC arrays, average Nusselt number asymptotes as the solid volume fraction approaches the close-packing limit of 0.732 in the FCC arrays for all values of Reynolds number considered. Comparison with the isolated Nusselt number correlation (dash-dotted line) shows that for  $\varepsilon_s = 0.01$  and  $Re > 50$ , the average Nusselt number is less than that for an isolated particle due to the shielding effects. This completes the discussion of ordered arrays. The remaining part of this section is devoted to random particle configurations in gas-solid flows.

### 4.1.2 Random arrays

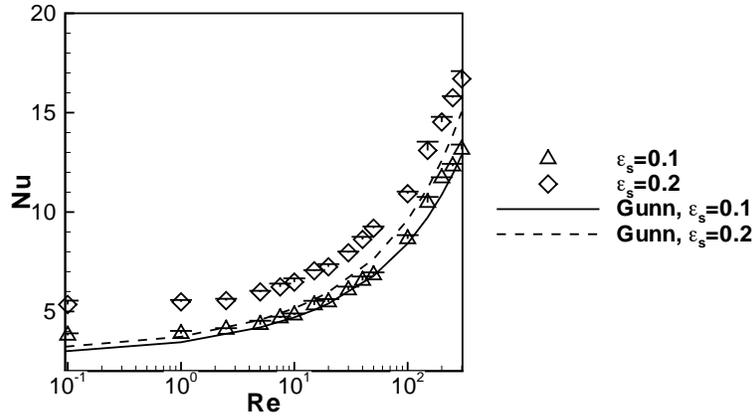


Figure 4.5 Comparison of the average Nusselt number in random arrays versus the Reynolds number  $Re$  for solid volume fractions equal to 0.1 and 0.2 from IBM simulations with Gunn's correlation.

Figure 4.5 shows the comparison of functional dependence of average Nusselt number  $Nu$  on Reynolds number for  $\varepsilon_s = 0.1$  and  $0.2$  obtained from IBM simulations with Gunn's correlation. For both solid volume fractions the maximum difference between IBM simulations and Gunn's correlation occurs in the Stokes flow regime. For example, at  $Re = 0.1$ , IBM simulations and Gunn's correlation differ by about 25% and 66% for  $\varepsilon_s = 0.1$  and  $0.2$ , respectively. As the Reynolds number increases, the difference between IBM simulations and Gunn's correlation also reduces. For  $\varepsilon_s = 0.1$ , Gunn's correlation and IBM simulations are in excellent agreement for  $Re > 1$ , while for  $\varepsilon_s = 0.2$ , reasonable agreement (within 20% difference) is observed for  $Re > 50$ . One principle difference between the IBM simulations and Gunn's correlation is the dependence of average Nusselt number on  $Re$  in the Stokes flow regime ( $Re < 1$ ). While the IBM simulations suggest an asymptotic value for the average Nusselt number in the Stokes flow regime, Gunn's correlation predicts a monotonic decrease in average Nusselt number with decreasing Reynolds number.

As shown in the previous chapter, the average Stokes Nusselt number  $Nu_0$  from scalar IBM simulations and Gunn's correlation differs by as much as 300% for the highest solid volume

fraction of 0.5 that is considered for random arrays in this study. Due to the increased difference in average Stokes Nusselt number with increasing solid volume fraction, it is natural to expect that range of Reynolds number for which the IBM simulations and Gunn's correlation will be in reasonable agreement (defined as the Reynolds number range for which difference is less than 20%) will also shrink with increasing solid volume fraction. For example, in Fig. 4.5 it is already observed that the range of reasonable agreement shrinks from  $Re > 1$  for  $\varepsilon_s = 0.1$  to  $Re > 50$  for  $\varepsilon_s = 0.2$ . Comparison of the dependence of average Nusselt number on Reynolds number from scalar IBM simulations with Gunn's correlation at even higher solid volume fractions of 0.3, 0.4, and 0.5 in Fig. 4.6 clearly reveals the shrinking of Re range with increasing solid volume fraction. For  $\varepsilon_s = 0.3$  and 0.4, scalar IBM simulations and Gunn's correlation are in reasonable agreement for  $Re > 100$  and  $Re > 225$ , respectively. Since the maximum Reynolds number considered for  $\varepsilon_s = 0.5$  in random arrays is equal to 100, the range of reasonable agreement is not apparent within this range of Reynolds number for  $\varepsilon_s = 0.5$ . But, nevertheless, in the range of reasonable agreement for all other volume fractions, it can be seen that the scalar IBM results and Gunn's correlation show a near identical trend with Reynolds number.

In the previous chapter 3, through a budget study of the scalar transport equation it was shown that the axial diffusion is one-third of the total diffusion over the entire range of solid volume fraction considered in the Stokes flow regime. The same budget study is extended to higher Reynolds numbers in the next section and it is shown that the axial diffusion remains one-third of the total diffusion, irrespective of the Reynolds number.

#### 4.1.3 Budget of the scalar transport equation

Decomposing the diffusion term  $\nabla^2\phi$  (see Chap. 3 for details) in the scalar transport equation as the sum of axial  $\widehat{D}_{\phi,\parallel}$  and perpendicular diffusion terms  $\widehat{D}_{\phi,\xi}$ , followed by defining local fluid-phase volumetric averages in the  $y-z$  plane along the mean flow direction  $x-$ , and taking the ensemble average results in three "ensemble averaged" local fluid-phase volumetric average terms for convection  $\langle \widehat{C}_{\phi}^{(f)} \rangle_{\mathcal{V}}(x)$ , axial diffusion  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}(x)$ , and perpendicular diffusion  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}(x)$ . The budget analysis of the scalar transport equation in the previous chapter

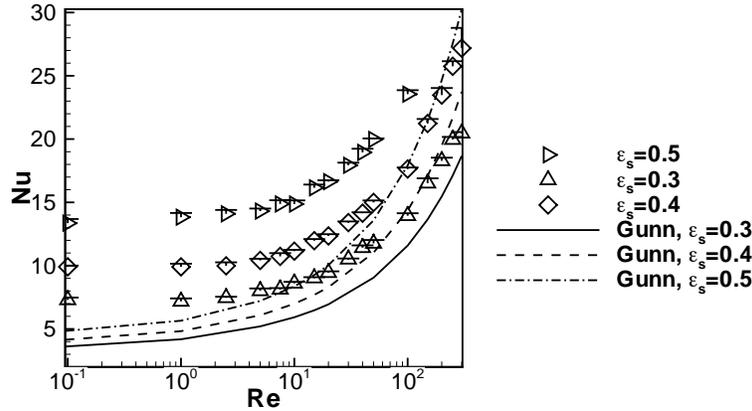


Figure 4.6 Comparison of the average Nusselt number in random arrays versus the Reynolds number  $Re$  for solid volume fractions equal to 0.3, 0.4, and 0.5 from IBM simulations with Gunn's correlation. Due to the restriction posed by the serial solver, high numerical resolutions required for  $\varepsilon_s = 0.5$  and  $Re > 100$  are not simulated.

revealed that axial diffusion was always one-third of the total diffusion for all solid volume fractions considered for both random and ordered arrays in the Stokes flow regime. Extending the budget study of scalar transport equation to finite Reynolds numbers is Fig. 4.7(a) which shows the comparison of relative magnitudes of the “ensemble averaged” local fluid-phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}$  (dash-dot line) along the  $x$ -axis for  $\varepsilon_s = 0.1$  and  $Re = 50$ . Figs. 4.7(b) shows the same comparison for a different  $(\varepsilon_s, Re)$  combination of  $(0.4, 50)$ . Two important observations can be made from these figures, which are discussed below in detail.

The first observation pertains to the contribution of axial diffusion term relative to the total diffusion. If the ratio of axial diffusion to total diffusion is computed as  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}} / (\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}} + \langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}})$ , then an average value of one-third is obtained for both for both cases shown in Fig. 4.7. Although not shown here, the above analysis when extended to even higher Reynolds numbers (up to  $Re = 300$ ) reveals a similar trend: axial diffusion remains approximately one-third of the total diffusion. Based on the current budget study and the one discussed in Chap. 3 for

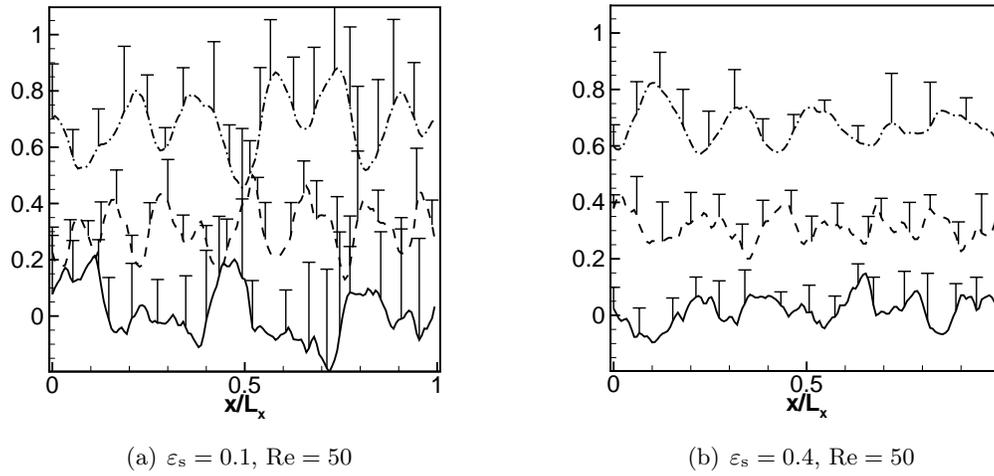


Figure 4.7 Comparison of relative magnitudes of the “ensemble averaged” local fluid-phase volumetric averages  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}$  (solid line),  $\langle \widehat{D}_{\phi,\parallel}^{(f)} \rangle_{\mathcal{V}}$  (dashed line), and  $\langle \widehat{D}_{\phi,\perp}^{(f)} \rangle_{\mathcal{V}}$  (dash-dotted line) along the  $x$ - axis obtained for the normalized convection  $\widehat{C}_\phi$ , axial diffusion  $\widehat{D}_{\phi,\parallel}$  and perpendicular diffusion terms  $\widehat{D}_{\phi,\perp}$ , respectively. Ensemble average has been performed over 5 independent realizations, and the scatter between the realizations is shown by the one-sided error bars.

the Stokes flow regime, it is concluded that axial diffusion is always one-third of the total diffusion, and that the axial diffusion should not be neglected in order to model fluid-phase heat transfer for any combination of  $\varepsilon_s$  and  $\text{Re}$ .

The second observation is related to the contribution of the convection term in the scalar transport equation. For the Stokes flow case, the convection term was found to be negligible in the previous chapter. However, for  $\text{Re} = 50$  in Figs. 4.7(a) and 4.7(b), it can be seen that the convection term becomes significant. Since the volume average of the convection term over the entire volume should be zero due to periodic boundary conditions, the local fluid-phase volumetric average  $\langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu(x)$  of the convection term fluctuates around zero such that  $\int_{\mathcal{L}_x} \langle \widehat{C}_\phi^{(f)} \rangle_{\mathcal{V}}^\mu(x) dx = 0$ .

## 4.2 A correlation for average Nusselt number $\text{Nu}$

From the scalar IBM simulation data for average Nusselt number for random arrays (Figs. 4.5 and 4.6), the correlation for average Nusselt number is extended to finite Reynolds numbers from the average Stokes Nusselt number  $\text{Nu}_0$  correlation proposed in Chap. 3. The new average Nusselt number correlation for  $\text{Pr} = 0.7$  is

$$\text{Nu} = \begin{cases} 2 + \frac{1}{(1-\varepsilon_s)^3} \left( 10.35\varepsilon_s + 5.51\varepsilon_s^2 - 18.16\varepsilon_s^{3/2} + 1.63\varepsilon_s^{1/3} \right) & \text{Re} < 1 \\ 2 \exp \left( 0.82 \varepsilon_s^{0.5} + 4.45 \varepsilon_s^{1.5} + 0.29 \text{Re}^{0.3} - 1.52 \varepsilon_s^{1.5} \text{Re}^{0.1} \right) & \text{Re} \geq 1. \end{cases} \quad (4.1)$$

## 4.3 Discussion and conclusions

Scalar IBM simulations have been extended from the Stokes flow regime in 3 to high Reynolds numbers up to  $\text{Re} = 300$ . Average Nusselt number as a function of Reynolds number and solid volume fraction is found from scalar IBM simulations for ordered and random arrays, and compared with the widely used heat and mass transfer correlation of Gunn.

For ordered arrays, the maximum difference for the average Nusselt number between scalar IBM simulations and Gunn's correlation is found in low (Stokes flow) to moderate Reynolds ( $\text{Re} < 50$ ) number range. As the Reynolds number increases, the difference between scalar

IBM results and Gunn's correlation first decrease and then increases again. This is due to the difference slopes for average Nusselt number in ordered arrays from scalar IBM simulations and Gunn's correlation, which is primarily for random arrays. Besides, due to the dependence of average Nusselt number on the mean flow angle for  $Re > 5$ , the comparison of scalar IBM results and Gunn's correlation for ordered arrays is only limited for  $Re > 5$ .

Comparison of local Nusselt number along the particle surface in SC arrays reveals that the maximum of local Nusselt number shifts from the regions of maximum flow channel width in Stokes flow regime toward the regions of minimum flow channel width. This is attributed to the dominance of convective transport over the diffusive transport as the Reynolds number increases. Since by mass continuity, the fluid velocity is highest in narrow regions between particles, in the convective transport dominated regime the maximum heat transfer, and the maximum local Nusselt number shift toward the regions of narrow flow channel widths.

For the lowest solid volume fraction of 0.1 for random arrays, scalar IBM simulations and Gunn's correlation are in excellent agreement for  $Re > 1$ . As the solid volume fraction increases in random arrays, the range of reasonable agreement (defined as the Reynolds range in which scalar IBM simulations and Gunn's correlation differ by less than 20%) also shrinks. For  $\varepsilon_s = 0.2, 0.3, \text{ and } 0.4$ , scalar IBM simulations and Gunn's correlation are in reasonable agreement for  $Re > 50, Re > 100, \text{ and } Re > 225$ , respectively. In the reasonable agreement range of Reynolds number for each solid volume fraction, the average Nusselt number from scalar IBM simulations and Gunn's correlation show a near identical trend with the Reynolds number. For both ordered and random arrays, the average Nusselt number from scalar IBM simulations remains nearly constant in the Stokes flow regime ( $Re \leq 1$ ). However, the average Nusselt number from Gunn's correlations shows a strong dependence on Reynolds number and decreases monotonically in the Stokes flow regime.

In the Stokes flow regime, Gunn's correlation was based on the conclusion drawn from his stochastic model that axial diffusion does not effect the Stokes Nusselt number. However, the budget study in chapter 3 revealed that axial diffusion is one-third of the total diffusion and its neglect will lead to under predictions for the average Stokes Nusselt number. For

higher Reynolds numbers, Gunn's correlation was based on data from the experiments (limited to only those experiments that considered axial diffusion at low Re) of Gunn and Desouza (1974), Turner and Otten (1973), and Denton (1951). For any solid volume fraction in random arrays, the maximum difference for average Nusselt number between the scalar IBM simulations and Gunn's correlation occur in the Stokes flow regime. As the solid volume fraction increases, the above difference also increases with the highest difference of 300% observed for  $\varepsilon_s = 0.5$ . As a result, the onset of range of reasonable agreement shrinks toward higher value of Reynolds number with increasing solid volume fraction.

The budget study of the scalar transport equation is extended from Stokes flow regime to high Reynolds numbers. It is found that axial diffusion term is one-third of the total diffusion for any combination of  $\varepsilon_s$  and Re. This is an important observation for researchers aiming to model heat transfer in the fluid-phase. A correlation for the average Nusselt number Nu as function of solid volume fraction  $\varepsilon_s$  and Reynolds number is proposed and recommended for use in simulations of gas-solid flows solving for average equations.

**CHAPTER 5. ACCURATE NUMERICAL ESTIMATION OF  
INTERPHASE MOMENTUM TRANSFER IN  
LAGRANGIAN–EULERIAN SIMULATIONS OF DISPERSED  
TWO–PHASE FLOWS**

This chapter is a printed manuscript (Garg et al., 2007) titled “Accurate numerical estimation of interphase momentum transfer in Lagrangian–Eulerian simulations of dispersed two–phase flows” in “Intl. J. Multiphase Flow” authored by R. Garg, C. Narayanan, D. Lakehal, and S. Subramaniam.

The Lagrangian–Eulerian (LE) approach is used in many computational methods to simulate two–way coupled dispersed two–phase flows. These include averaged equation solvers, as well as direct numerical simulations (DNS) and large–eddy simulations (LES) that approximate the dispersed–phase particles (or droplets or bubbles) as point sources. Accurate calculation of the interphase momentum transfer term in LE simulations is crucial for predicting qualitatively correct physical behavior, as well as for quantitative comparison with experiments. Numerical error in the interphase momentum transfer calculation arises from both forward interpolation/approximation of fluid velocity at grid nodes to particle locations, and from backward estimation of the interphase momentum transfer term at particle locations to grid nodes. A novel test that admits an analytical form for the interphase momentum transfer term is devised to test the accuracy of the following numerical schemes: (1) fourth–order Lagrange Polynomial Interpolation (LPI-4), (2) Piecewise Cubic Approximation (PCA), (3) second–order Lagrange Polynomial Interpolation (LPI-2) which is basically linear interpolation, and (4) a Two–Stage Estimation algorithm (TSE). A number of tests are performed to systematically characterize the effects of varying the particle velocity variance, the distribution of particle positions, and

fluid velocity field spectrum on estimation of the mean interphase momentum transfer term. Numerical error resulting from backward estimation is decomposed into statistical and deterministic (bias and discretization) components, and their convergence with number of particles and grid resolution is characterized. It is found that when the interphase momentum transfer is computed using values for these numerical parameters typically encountered in the literature, it can incur errors as high as 80% for the LPI-4 scheme, whereas TSE incurs a maximum error of 20%. The tests reveal that using multiple independent simulations and higher number of particles per cell are required for accurate estimation using current algorithms. The study motivates further testing of LE numerical methods, and the development of better algorithms for computing interphase transfer terms.

## 5.1 Introduction

The Lagrangian–Eulerian(LE) approach is widely used to simulate dispersed two-phase flows. This work focuses on the development of accurate numerical methods for computing the interphase momentum exchange term in LE simulations of two-phase flows with non-negligible mass loading. Therefore, the findings of this study are relevant to two-phase flows that must account for two-way coupling. Numerical error incurred in estimating the interphase momentum transfer term directly affects the fluid velocity solution, and feeds back to the particle trajectories. These errors can drastically affect the physical picture that emerges from an LE simulation. The conclusions of this study can also be easily generalized to the mass and energy interphase exchange terms.

### 5.1.1 Physical system

In the LE approach the dispersed–phase consisting of  $N_p$  physical particles<sup>1</sup> is represented in a Lagrangian frame at time  $t$  by  $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t), i = 1, \dots, N_p(t)\}$ , where  $\mathbf{X}^{(i)}(t)$  denotes the  $i^{th}$  particle’s position and  $\mathbf{V}^{(i)}(t)$  represents its velocity. For the sake of simplicity we consider monodisperse particles here, although the conclusions of this work hold for polydisperse systems

<sup>1</sup>By particle we mean any dispersed–phase element, including solid particles, droplets and bubbles.

also. For monodisperse particles with diameter  $D_p$ , the particle mass is the same for each particle  $m^{(i)} = m_p = \rho_p V_p$ , where  $\rho_p$  and  $V_p = \pi D_p^3/6$  are the individual particle density and volume respectively. The position and velocity of the physical particles evolve by

$$\frac{d\mathbf{X}^{(i)}}{dt} = \mathbf{V}^{(i)} \quad (5.1)$$

$$m_p \frac{d\mathbf{V}^{(i)}}{dt} = \mathbf{f}^{(i)}, \quad i = 1, \dots, N_p(t) \quad (5.2)$$

where  $\mathbf{f}^{(i)}$  is the instantaneous force acting on the  $i^{\text{th}}$  physical particle.

For the case of volumetrically dilute flows<sup>2</sup> with finite mass loading, the momentum conservation in the fluid phase is the single-phase momentum conservation equation augmented by an interphase momentum transfer term  $\mathbf{F}^{fp}$ , which accounts for the coupling of the dispersed-phase momentum with the fluid phase:

$$\rho_f \left( \frac{\partial \mathbf{U}^f}{\partial t} + \mathbf{U}^f \cdot \nabla \mathbf{U}^f \right) = \nabla \cdot \boldsymbol{\tau} - \mathbf{F}^{fp}. \quad (5.3)$$

This general formulation of the LE approach subsumes the application of the LE method to dispersed two-phase flows in three different simulation contexts: (1) direct numerical simulation (DNS) using a point-particle approximation for the dispersed phase, (2) large eddy simulation (LES), and (3) computational fluid dynamics (CFD) using averaged equations for the carrier flow. The specific equations appropriate to each of these simulation methods can be recovered by appropriate interpretation (realization, filtered realization or statistical average) of the fluid velocity field, stress tensor and interphase momentum transfer term. Table 5.1 lists the representation of the carrier flow field and dispersed phase for these three simulation methods. This paper focuses primarily on accurate estimation of the interphase momentum transfer term  $\mathbf{F}^{fp}(\mathbf{x}, t)$  in the context of CFD, where both fluid and particle phases are represented in a statistically averaged sense. However, the conclusions of this paper are equally applicable and relevant to the hybrid simulations DNS<sup>(b)</sup> and LES<sup>(b)</sup> in Table 5.1.

The equation for conservation of mean momentum in the fluid phase is obtained by ensemble

<sup>2</sup>This assumption does not pose an inherent limitation on our investigation, but we choose this case to simplify the equations. The conclusions of this work will also hold for non-dilute cases but volume displacement effects will need to be accounted for.

Simulation Method	Carrier flow fields: $\mathbf{U}^f(\mathbf{x}, t), p(\mathbf{x}, t)$	Dispersed phase: $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t), i = 1, \dots, N_p(t)\}$
DNS <sup>(a)</sup> with physical particles	Realization	Realization: point field
DNS <sup>(b)</sup> with stochastic particles	Realization	Statistically averaged density
LES <sup>(a)</sup> with physical particles	Filtered field of a realization	Spatially filtered point field
LES <sup>(b)</sup> with stochastic particles	Filtered field of a realization	Spatially filtered density
CFD	Mean fields	Statistically averaged density

Table 5.1 Representation of carrier flow and dispersed phase in different LE simulations: DNS<sup>(b)</sup> and LES<sup>(b)</sup> are denoted hybrid simulations.

averaging (Drew and Passman, 1998)

$$\rho_f \alpha_f \left( \frac{\partial \langle \mathbf{U}^f \rangle}{\partial t} + \langle \mathbf{U}^f \rangle \cdot \nabla \langle \mathbf{U}^f \rangle \right) = \nabla \cdot \langle \boldsymbol{\tau} \rangle - \langle \mathbf{F}^{fp} \rangle + \boldsymbol{\tau}^{RS}, \quad (5.4)$$

where  $\alpha_f$  is the average fluid volume fraction,  $\rho_f$  is the thermodynamic density of the fluid phase (assumed constant),  $\boldsymbol{\tau}^{RS}$  is the residual stress resulting from ensemble averaging, and the angle brackets represent phasic averages of the terms.

Based on a statistical representation of the dispersed phase as a point process (Subramaniam, 2000, 2001) one can associate a density  $f(\mathbf{x}, \mathbf{v}, t)$  with the ensemble of realizations  $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t), i = 1, \dots, N_p(t)\}$ . The density  $f(\mathbf{x}, \mathbf{v}, t)$  admits a decomposition

$$f(\mathbf{x}, \mathbf{v}, t) = n(\mathbf{x}, t) f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}; t), \quad (5.5)$$

where  $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}; t)$  is the pdf of particle velocity conditional on physical space and  $n(\mathbf{x}, t)$  is the density of expected number of particles in physical space. In this notation  $\mathbf{v}$  is the sample space variable corresponding to particle velocity  $\mathbf{V}$ . The expected value of the interphase momentum transfer term (or fluid-particle interaction force)  $\mathbf{F}^{fp}(\mathbf{x}, t)$  can be written as an integral over velocity space:

$$\langle \mathbf{F}^{fp} \rangle(\mathbf{x}, t) = \int_{[\mathbf{v}]} \langle \mathbf{f} | \mathbf{x}, \mathbf{v}; t \rangle n(\mathbf{x}, t) f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}; t) d\mathbf{v}, \quad (5.6)$$

where  $\langle \mathbf{f} \mid \mathbf{x}, \mathbf{v}; t \rangle$  is the conditional average of the force acting on the physical particles. It is this quantity that we seek to calculate accurately in our study. Since we only refer to average fluid velocity and average interphase momentum transfer from here on, to improve readability the angle bracket notation is omitted from these quantities in the rest of the paper.

### 5.1.2 Computational representation

In LE simulations the dispersed-phase density  $f(\mathbf{x}, \mathbf{v}, t)$  is indirectly represented by  $N_c$  computational particles at time  $t$  in a Lagrangian frame by  $\{\mathbf{X}^{*(i)}(t), \mathbf{V}^{*(i)}(t), i = 1, \dots, N_c(t)\}$ , with  $\mathbf{X}^{*(i)}(t)$  denoting the  $i^{\text{th}}$  computational particle's position and  $\mathbf{V}^{*(i)}(t)$  its velocity. The number of computational particles  $N_c$  does not necessarily have to equal the number of physical particles  $N_p$ , which in our point process model is a random number. Typically  $N_c$  is chosen to be smaller than  $N_p$  by even orders of magnitude sometimes, and the correspondence between the computational representation and the physical system is enforced in the following statistical sense.

The number of physical particles represented by the  $i^{\text{th}}$  computational particle is denoted by  $n_p^{(i)}$ , such that the sum over all the computational particles is equal to the expected number of physical particles

$$\sum_{i=1}^{N_c} n_p^{(i)} = \langle N_p \rangle. \quad (5.7)$$

Therefore, the statistical weight assigned to each computational particle is

$$\mu^{(i)} = \frac{n_p^{(i)}}{\sum_{i=1}^{N_c} n_p^{(i)}} = \frac{n_p^{(i)}}{\langle N_p \rangle}, \quad (5.8)$$

which satisfies the normalization property  $\sum_{i=1}^{N_c} \mu^{(i)} = 1$ .

The position and velocity of each computational particle evolve by the equations

$$\frac{d\mathbf{X}^{*(i)}}{dt} = \mathbf{V}^{*(i)} \quad (5.9)$$

$$m_p \frac{d\mathbf{V}^{*(i)}}{dt} = \mathbf{f}^{*(i)}, \quad i = 1, \dots, N_c(t), \quad (5.10)$$

where  $\mathbf{f}^{*(i)}$  is the *modeled* force acting on the  $i^{\text{th}}$  computational particle. The computational particle position  $\mathbf{X}^{*(i)}$  and velocity  $\mathbf{V}^{*(i)}$  are evolved in time from initial conditions at time  $t_0$

that correspond to a specified initial number density  $n(\mathbf{x}, t_0)$  and velocity probability density function  $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}; t_0)$  of the physical particles.

A general form of the particle force model that subsumes different drag force correlations is:

$$\mathbf{f}^{*(i)}(t) = \mathbf{f} \left( \mathbf{U}^f \left( \mathbf{X}^{*(i)}(t), t \right), \mathbf{V}^{*(i)}, \rho_f, \nu_f, \rho_p, D_p \right), \quad (5.11)$$

where  $\rho_f$  and  $\nu_f$  is the fluid phase density and kinematic viscosity, respectively. A more general force model could include additional terms such as the added mass term, Basset history term, or Saffman lift, as dictated by the problem physics. Even though we only model the drag in our study, our conclusions regarding the accurate numerical calculation of the interphase momentum transfer term will apply to this wider class of flows, with minor modifications to account for the changes in the functional form  $\mathbf{f}$  that will be necessitated by the additional physics.

### 5.1.3 Problem Statement

Proper representation of the flow physics in an LE simulation is contingent upon accurate calculation of the mean interphase momentum transfer term  $\mathbf{F}_{fp}(\mathbf{x}, t)$  from the LE solution, i.e., the mean fluid velocity field  $\mathbf{U}^f(\mathbf{x}, t)$ , and the position and velocity of the computational particles  $\{\mathbf{X}^{*(i)}(t), \mathbf{V}^{*(i)}(t), i = 1, \dots, N_c\}$ . The mean interphase momentum transfer term  $\mathbf{F}_{fp}(\mathbf{x}, t)$  at Eulerian grid nodes is estimated from this solution data in two steps:

1. *Calculation of particle forces  $\mathbf{f}^{*(i)}$ :*

This requires calculation of the fluid velocity at the particle location  $\mathbf{U}^f(\mathbf{X}^{*(i)}, t)$  in 5.11 from the fluid velocity at Eulerian grid nodes. The numerical estimate of the fluid velocity field  $\mathbf{U}^f(\mathbf{x}, t)$  at the particle location  $\mathbf{X}^{*(i)}$  using a representation of  $\mathbf{U}^f$  at  $M$  grid nodes is denoted  $\{\mathbf{U}^f(\mathbf{X}^{*(i)}, t)\}_M$ , and is obtained through *forward interpolation/approximation* as:

$$\{\mathbf{U}^f(\mathbf{X}^{*(i)}, t)\}_M = \mathcal{F} \left\{ \mathbf{U}_m^f, m = 1, \dots, M; \mathbf{X}^{*(i)} \right\}, \quad (5.12)$$

where the fluid velocity at the  $m^{\text{th}}$  Eulerian grid node is denoted  $\mathbf{U}_m^f$ , and  $\mathcal{F}$  is a generic

interpolation/approximation operation. The particle force  $\mathbf{f}^{*(i)}$  is then obtained by substituting  $\{\mathbf{U}^f(\mathbf{X}^{*(i)}, t)\}_M$  for  $\mathbf{U}^f(\mathbf{X}^{*(i)}, t)$  in 5.11.

2. *Mean interphase momentum transfer  $\mathbf{F}_{fp}(\mathbf{x}, t)$  from particle forces  $\mathbf{f}^{*(i)}$ :*

The numerical procedure to calculate the Eulerian mean field  $\mathbf{F}_{fp}(\mathbf{x}, t)$  from particle data  $\{\mathbf{X}^{*(i)}(t), \mathbf{f}^{*(i)}(t), i = 1, \dots, N_c\}$  is describe variously as mean estimation from particle data, projection of fluid–particle interaction forces onto the Eulerian grid, or backward estimation. The numerical estimate for the mean interphase momentum transfer  $\mathbf{F}^{fp}(\mathbf{x}, t)$  at the  $m^{\text{th}}$  Eulerian grid node is denoted  $\{\mathbf{F}_m^{fp}\}$ , and the general form of its estimate from the particle data is:

$$\{\mathbf{F}_m^{fp}\} = \mathcal{E} \left\{ \mathbf{X}^{*(i)}, \mathbf{f}^{*(i)}, n_p^{(i)}, i = 1, \dots, N_c(t) \right\}, \quad (5.13)$$

where  $\mathcal{E}$  like  $\mathcal{F}$  is another generic interpolation/approximation operator.

#### 5.1.4 Review of existing schemes

Both forward interpolation and the estimation of mean fields from particle data have been studied by other researchers, and a selective review that motivates this study follows.

##### 5.1.4.1 Forward Interpolation

Yueng and Pope (1988) investigated many numerical approaches for interpolation of fluid velocity at a Lagrangian particle location in homogeneous turbulence. Among the schemes they considered are a trilinear scheme, a 13-point third-order scheme based on Taylor series (TS–13), and a fourth-order cubic spline. Their study shows that the fourth-order spline is most accurate for forward interpolation, followed by the TS–13 scheme. The trilinear interpolation scheme was found to be unacceptably poor. Balachandar and Maxey (1989) also analyzed various numerical schemes to calculate the fluid velocity at a particle location in one-way coupled spectral simulations of decaying homogeneous turbulence by comparing them with the most accurate, and also the most computationally expensive, direct summation (DS) scheme. They studied the TS–13 scheme, sixth-order Lagrangian interpolation (LPI-6), partial Hermite

interpolation (PHI), shape function method (SFM), and linear interpolation (LPI-2). They find that the estimation of statistical quantities such as Lagrangian velocity correlation, effective eddy diffusivity, and mean square particle dispersion are not sensitive to the approximation scheme used. They show that on a  $32^3$  grid size at low Reynolds number ( $Re_\lambda = 17$ ) the LPI-6 scheme is sufficient to accurately extract quantities such as absolute velocity of single particle and also the relative velocity of two particles. However, at higher Reynolds number ( $Re_\lambda = 26.5$ ) the more accurate PHI scheme is needed, at additional computational expense. It should be noted that the TS-13 scheme is best suited to simulations of homogeneous turbulence that incorporate a de-aliasing procedure and make use of staggered grid. The PHI and SFM schemes have been developed specifically for spectral simulation. While these studies provide useful guidelines to choose appropriate schemes for forward interpolation, they only address the first step in accurate estimation of the mean interphase momentum transfer term.

#### 5.1.4.2 Estimation of mean field from particle data

Various approaches have been proposed for the second step that involves estimation of the mean interphase momentum transfer term  $\mathbf{F}_m^{fp}$  from particle data  $\{\mathbf{X}^{*(i)}(t), \mathbf{f}^{*(i)}(t), i = 1, \dots, N_c\}$ . We review three principal approaches here: (1) the particle-in-cell (PIC) method, (2) the projection onto neighboring nodes (PNN) method, and (3) the projection onto identical stencil (PIS) method.

*PIC:* Crowe (1982) extended the particle-in-cell (PIC) method (Evans and Harlow, 1957; Harlow, 1988) to calculate the mean interphase momentum transfer term. In this method, the mean interphase momentum transfer term is calculated as the summation of forces  $\mathbf{f}^{*(i)}$  exerted on the fluid by each particle in the control volume surrounding a grid node as shown in Figure 5.1(a) and also expressed by Eq. B.1. This is effectively a box kernel, which has the disadvantage that its estimate is piecewise-constant in physical space (Pope, 2000). Therefore this method cannot be used to calculate gradients of the mean interphase momentum transfer field, if they are needed.

*PNN*: This is a so-called projection method wherein rather than summing all the particle forces around a grid node, each particle force is projected onto the neighboring grid points (eight in 3-D, four in 2-D) based on a weighting scheme (see Fig. 5.1(b)). The weights can be based on the cell volumes as in Squires and Eaton (1990), or on the distance between the particle and the node as used by Elghobashi and Truesdell (1993). The expression for estimation by PNN method based on the distance between particle and the node is given by Eq. B.3.

Boivin et al. (1998) compared PIC and PNN methods by first forward interpolating a turbulent velocity field specified at Eulerian grid nodes to randomly generated off-grid particle locations using a third-order LPI scheme. These interpolated fluid velocities at the particle locations are then used as particle data to estimate the fluid velocity field at the Eulerian grid nodes using PIC and PNN. Their test results show that the PNN scheme results in a fluid velocity spectrum that is closer to the original velocity field spectrum field than that obtained using the PIC method. However, the conclusions of the Boivin study need to be interpreted carefully because their test is significantly different from our problem of mean interphase momentum transfer estimation in many respects. Since the Boivin et al. (1998) study only tries to recover the fluid velocity field instead of the interphase momentum transfer term (cf. Eq. 5.11), it is not affected by the particle velocity  $\mathbf{V}^{*(i)}$  or its pdf  $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}; t)$ . Boivin et al. randomly assign only one particle location to each cell. As is shown later in this paper, this results in an unacceptably high level of statistical error.

*PIS*: Sundaram and Collins (1996) show that in order to ensure overall energy balance, the order of the interpolation scheme used in the forward interpolation should be the same as that used in the backward estimation. We noted earlier in this paper that the studies on forward interpolation reveal that at least for turbulent velocity fields in DNS, high-order schemes like TS-13 or LPI-6 are needed for accuracy. These high-order schemes have broad stencils in physical space that extend well beyond the cell where the particle is present. A PIS scheme will then require a weighting kernel with identically broad support to compute the mean interphase momentum transfer from particle forces. Each particle exerts a non-local influence

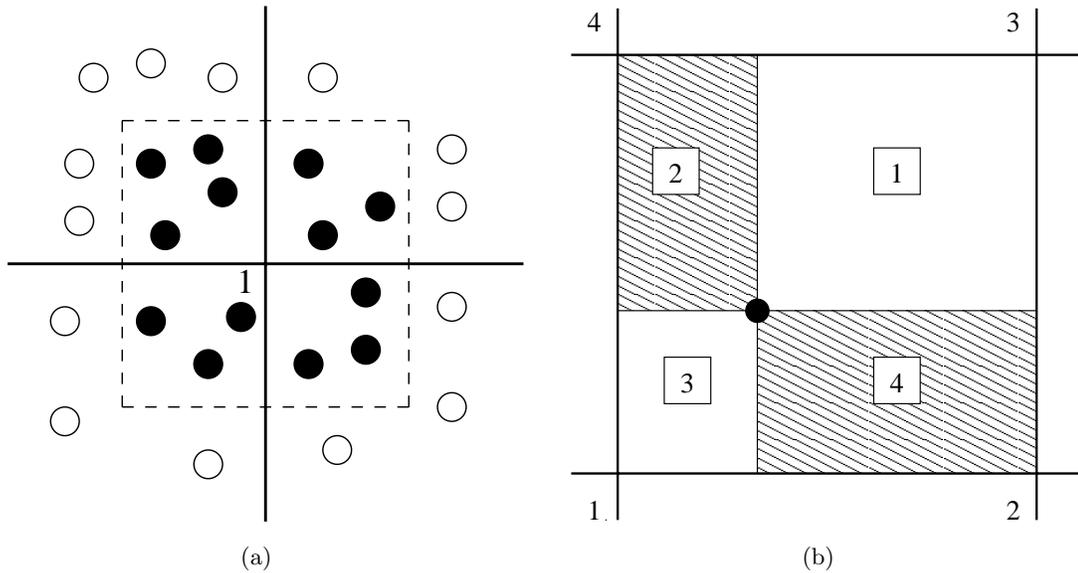


Figure 5.1 Sketch showing the PIC and PNN mean estimation schemes: (a) Mean estimation by PIC method in 2-D. Grid node 1 receives the full contribution from particles (shown as black spheres) located in cell area (shown by dotted lines) around it. (b) Mean estimation by PNN method in 2-D. For a particle (shown as a black sphere) in a two-dimensional cell, grid node 1 receives a fraction of the particle force which is proportional to the area of region 1 divided by the entire cell area.

on the estimate of the interphase momentum transfer, and this raises a concern whether the numerics is consistent with the flow physics. Using a fourth-order LPI scheme for both forward interpolation and backward estimation (in Eqs. B.2 and B.3, respectively) on coarse and fine grid resolution, they note that the result of spreading particle influence over a large volume does not significantly affect the dynamics of the mean energy in a particle-laden turbulent flow. On this basis, Sundaram and Collins (1996) assert that the PIS symmetry in the order of the scheme used for both forward interpolation and backward estimation is required, even if it increases the domain of influence of each particle due to a broad interpolation stencil.

Narayanan et al. (2002) assess the relative merits of the PNN and PIS methods by comparing the growth rates of mixing layers obtained using LE simulations with these schemes, to those obtained from a linear stability analysis. However, the results obtained for growth rates are too close to draw any conclusions about the relative merits of the two methods.

In all LE numerical implementations, including those cited above, there are two numerical parameters: the number of Eulerian grid cells and the number of computational particles. The estimate for the mean interphase momentum transfer term  $\mathbf{F}^{fp}$  on an Eulerian grid is obtained from a finite number of particle forces  $\mathbf{f}^*$ . This leads to *statistical error*, which can only be eliminated in the limit of infinite particles (also called the dense data limit). This limit is only asymptotically approached by simulations with a very large number of particles, and such calculations are expensive. Typical LE simulations must be reasonably accurate in a range of finite number of particles. A finite number of grid cells also leads to *spatial discretization error* as in CFD of single-phase flow. Numerical schemes in the LE context need to balance statistical and spatial discretization error.

In spite of the considerable work on forward interpolation as well as projection methods (PIC, PNN, PIS), there is no comprehensive study that quantifies the spatial and statistical error resulting from numerical estimation of mean interphase momentum transfer. The conclusions of Boivin et al. (1998) are based on a single test with  $96^3$  particles that does not quantify the statistical error, or its scaling with the number of particles. The Sundaram and Collins (1996) study tests only the fourth-order LPI scheme and does not quantify spatial and statistical error. Narayanan et al. (2002) consider LPI schemes of different orders but they do not characterize the behavior of spatial or statistical error. Lakehal and Narayanan (2003) quantify the effect of varying the total number of particles in an LE simulation on calculation of the average interfacial force. They find that increasing the number of particles shows a reduction in statistical noise, and the estimated interfacial force tends to an asymptotic value. However, this study also does not decompose the error into deterministic and statistical components. Also while numerical convergence with number of particles is empirically demonstrated, the accuracy of the scheme is not quantified. Are et al. (2005) investigate only spatial discretization error by considering the limit of dense data (1 billion particles).

In this work we construct a test problem for which the interphase momentum transfer term can be calculated analytically. We then compare the numerical error incurred by four different schemes in estimation of the mean interphase momentum transfer term. The total numerical

error is decomposed into statistical and deterministic components. Statistical error is defined as the fluctuations in interphase force estimation that arise as a result of finite particles. Deterministic error, which is further decomposed into bias and discretization components, results from finite number of particles and grid size, respectively. The individual contributions to the total numerical error from finite number of particles (statistical error and bias error) and finite grid size (spatial discretization error) are identified. The behavior of statistical error, bias error, and spatial discretization error is characterized over a range of grid sizes and total number of particles.

The four numerical schemes for calculation of the mean interphase momentum transfer term that are considered in this work are:

- (1) *LPI-4*: This is a fourth-order Lagrange polynomial interpolation (LPI) which has been widely used for both forward and reverse interpolation (Sundaram and Collins, 1996; Narayanan et al., 2002; Sundaram and Collins, 1999). It is a true interpolation scheme because it recovers the specified values of fluid velocity at grid nodes. The LPI-4 basis functions are shown in Fig. 5.2(a). Since this scheme is fourth-order accurate (Conte and Boor, 1980), in forward interpolation the error incurred using LPI-4 should exhibit fourth-order convergence with respect to grid spacing for a uniform grid. The LPI-4 stencil is four grid cells wide, as shown in Fig. 5.2(a). In backward estimation also its kernel bandwidth is four grid cells wide. The kernel bandwidth determines the extent to which Lagrangian particle data is smeared on the Eulerian flow grid in backward estimation.
- (2) *PCA*: This scheme has piecewise continuous cubic polynomial basis functions that are similar to the kernel derived by Monaghan and Lattanzio (1985) based on B-spline functions. See Fig. 5.2(b) for the PCA basis functions. It is important to note that this is not standard cubic spline interpolation that involves a matrix solution for the spline coefficients. In fact, this is only a piecewise cubic *approximation* that does not exactly recover specified values of the velocity field at the grid nodes. To distinguish it from the standard cubic spline interpolation, this scheme is referred to as piecewise cubic approx-

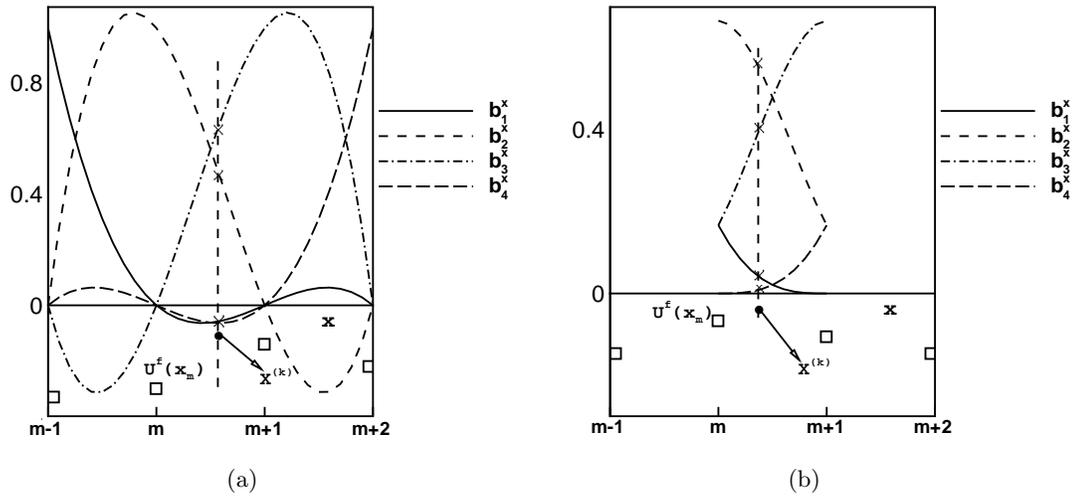


Figure 5.2 (a) Basis functions for LPI-4. (b) Basis functions for PCA. In both figures, squares represent the fluid velocity at that grid node,  $X^{(k)}$  is the location of particle (shown by black sphere) located between nodes  $m$  and  $m + 1$ . The intersections of the vertical dashed line with the curves (shown by crosses) indicates the value of the basis function at  $X^{(k)}$  that multiplies the nodal fluid velocity in Eq. B.2 to compute the fluid velocity at  $X^{(k)}$ .

imation (PCA). Monaghan (1992) notes that this scheme is only second-order accurate, in contrast to cubic spline interpolation which is fourth-order accurate. In backward estimation its kernel bandwidth is four grid cells wide.

- (2) *LPI-2*: This is a second-order Lagrange polynomial interpolation scheme, which is essentially a trilinear interpolation scheme that is identical to the PNN method (Squires and Eaton, 1990; Elghobashi and Truesdell, 1993; Boivin et al., 1998). It is a true interpolation scheme that is formally second-order accurate for forward interpolation. In backward estimation its kernel bandwidth is two grid cells wide.
- (4) *TSE*: This two-stage estimation algorithm is developed by Dreeben and Pope (1992). It is useful in simulations that involve unstructured meshes (Subramaniam and Haworth, 2000). For forward interpolation it is identical to LPI-2, and is formally second-order accurate. For backward estimation it employs a grid-free two-stage algorithm. In the first stage, it estimates weighted values of the particle property using a linear kernel of

user-specified bandwidth (e.g., interphase force) at knot locations that depend on where the particles are located. The second stage involves least-squares fitting of locally linear or quadratic functions to these knot values. The advantage of this method is that its convergence characteristics are not tied to the Eulerian grid (in fact it does not need an Eulerian grid at all!), but by adjusting the bandwidth of the kernel the user can balance the contribution from truncation and statistical errors.

For complete details of the interpolation schemes, the reader is referred to Appendix B.

## 5.2 Test Problem

We consider a simple physical problem to examine the numerical convergence and accuracy of the four schemes in calculating the mean interphase momentum transfer term. 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 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 force  $\mathbf{f}^{*(i)}$  is given by

$$\mathbf{f}^{*(i)} = m_p \frac{\mathbf{U}^f(\mathbf{X}^{*(i)}) - \mathbf{V}^{*(i)}}{\tau_p}, \quad (5.14)$$

where  $\tau_p = \rho_p D_p^2 / (18\nu_f \rho_f)$  is the particle momentum response time. In this test we do not consider time evolution, but simply evaluate the mean interphase momentum transfer term at some fixed time instant  $t$ . Therefore the time dependence is omitted in the rest of the description of this static test.

We consider a statistically homogeneous problem where the particle velocity distribution is independent of physical location  $\mathbf{x}$ , so that  $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x}) = f_{\mathbf{V}}(\mathbf{v})$ . If the particle density in physical space  $f_{\mathbf{X}}(\mathbf{x}) = n(\mathbf{x})/\langle N_p \rangle$  is known, then Eq. 5.6 simplifies to

$$\langle \mathbf{F}^{fp} \rangle(\mathbf{x}) = \langle N_p \rangle \int_{[\mathbf{v}]} \langle \mathbf{f} | \mathbf{x}, \mathbf{v} \rangle f_{\mathbf{X}}(\mathbf{x}) f_{\mathbf{V}}(\mathbf{v}) d\mathbf{v}. \quad (5.15)$$

If the mean fluid velocity field  $\mathbf{U}^f(\mathbf{x})$  is specified, along with the particle position and velocity distributions, the final analytical expression for  $\langle \mathbf{F}^{fp} \rangle$  from Eq. 5.15 is:

$$\langle \mathbf{F}^{fp} \rangle(\mathbf{x}) = \frac{m_p \langle N_p \rangle}{\tau_p} \left[ \mathbf{U}^f(\mathbf{x}) f_{\mathbf{x}}(\mathbf{x}) - \langle \mathbf{V} \rangle f_{\mathbf{x}}(\mathbf{x}) \right]. \quad (5.16)$$

It is interesting to note that although in the above equation  $\langle \mathbf{F}^{fp} \rangle$  is independent of the variance in particle velocity, numerical estimates for this quantity suffer from statistical noise which increases with particle velocity variance.

The estimate of mean interphase momentum transfer term depends on (i) the mean fluid velocity field, (ii) the particle position distribution, and (iii) the particle velocity distribution<sup>3</sup>. The following specification of the mean fluid velocity field, and the particle position and velocity distribution define the baseline test case, which we denote Test 1. The fluid velocity field  $\mathbf{U}^f = \{U_1^f, 0, 0\}$  is chosen to be of a simple transcendental form

$$U_1^f(x, y) = \cos\left(\frac{2\pi x}{\mathcal{L}_x}\right) \cos\left(\frac{2\pi y}{\mathcal{L}_y}\right), \quad (5.17)$$

in a domain  $\mathcal{D} = [0, \mathcal{L}_x] \times [0, \mathcal{L}_y] \times [0, \mathcal{L}_z]$ . The particle positions are randomly chosen according to a uniform distribution in the domain  $\mathcal{D}$ . The particle velocity  $\mathbf{V} = \{V_1, 0, 0\}$  is specified by the distribution of  $V_1$ , which is chosen to be a Gaussian with unit mean and variance  $\sigma^2$ . For the baseline test the variance is chosen to be zero, which corresponds to a delta-function specification of the particle velocity distribution. Figure 5.3 shows the contour plot of scaled analytical mean interphase momentum transfer term in the  $x$ -direction obtained from Eq. 5.16 for the baseline test case. This baseline test case is used to completely characterize the statistical error, bias error, and spatial discretization error for the four numerical schemes over a wide range of numerical parameter values.

We consider three variants of the baseline case in our tests to specifically probe certain other convergence characteristics of the numerical schemes used to estimate the mean interphase momentum transfer term. Unless noted otherwise, the problem parameters are retained at their baseline values. In the first variant (Test 2), a nonzero particle velocity variance is introduced

<sup>3</sup>Although the analytical value depends only on the mean particle velocity, the numerical estimate depends on the variance of the particle velocity distribution.

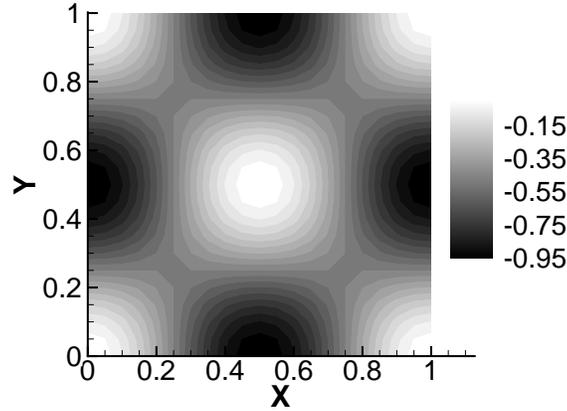


Figure 5.3 Contour plot of scaled analytical mean interphase momentum transfer term  $\langle F_x^{fp} \rangle / \langle F_x^{fp} \rangle_{\max}$  for the fluid velocity field given by Eq. 5.17 and mean particle velocity  $\langle V_1 \rangle = 1.0$ .

to represent nonzero particle velocity fluctuations that can be expected in most practical two-phase flows. For the linear drag law considered here, this nonzero particle velocity variance manifests itself as statistical noise in the estimate of the mean interphase momentum transfer. This test assesses the capability of the various schemes to yield accurate estimates of the mean interphase momentum transfer term with finite computational particles for noisy data.

Tests 3 and 4 are variants of Test 2 that consider the effect of changing the particle position distribution and spectrum of the fluid velocity field, respectively. In these tests we only characterize the total error, but we do not identify individual contributions. In the Test 3 we investigate the effect of a nonuniform distribution of physical particles while retaining the nonzero particle velocity variance in Test 2. If we do not introduce any computational particle number density control, the distribution of computational particles mimics that of the physical particles and we essentially generate nonuniform sampling. This test is representative of the spatial inhomogeneity in number density that is encountered in LE computations of real two-phase flows.

In Test 4, the effect of changing the spectrum of the fluid velocity field is investigated by

changing the wavelength of the cosine waves in Eq. 5.17 (the nonzero particle velocity variance of Test 2 is retained). Our intent in performing this test is to characterize the applicability of the four schemes to CFD, LES<sup>(b)</sup> and DNS<sup>(b)</sup>, each of which has progressively more high-wavenumber content in the velocity field. By changing the wavelength of the cosine waves on a fixed grid, we effectively vary the resolution of the velocity field, and investigate the consequences on the computed interphase momentum transfer term.

### 5.3 Numerical Analysis

In order to calculate a numerical estimate of  $\langle \mathbf{F}^{fp} \rangle(\mathbf{x})$ , the physical domain  $\mathcal{D}$  is discretized using a structured grid with  $M_x \times M_y \times M_z$  cells. In all our tests the domain is a unit cube with  $10 \leq M_x = M_y \leq 60$ , and  $M_z = 3$ . Since the mean velocity field is only a function of  $(x, y)$ , we use more grid cells in the  $x$ - $y$  plane. The expected total number of physical particles  $\langle N_p \rangle$  is represented by  $N_c$  computational particles, with each computational particle representing  $n_p = \langle N_p \rangle / N_c$  physical particles, resulting in equal statistical weight  $\mu = 1/N_c$  for each computational particle. The average number of computational particles in a grid cell is denoted  $N_{pc} = N_c/M$ , where  $M = M_x M_y M_z$  is the total number of grid cells. The numerical parameters affecting the accuracy of mean interphase momentum transfer term estimation are (i) the number of computational particles per cell  $N_{pc}$ , and (ii) grid size, which we represent by total number of nodes  $M$ .

The numerical estimate for  $\mathbf{F}^{fp}(\mathbf{x})$  at the  $m^{\text{th}}$  grid node obtained from the above discretization ( $M$  cells and  $N_{pc}$  particles per cell) is written as

$$\{\mathbf{F}_m^{fp}\}_{N_{pc}, M} = \frac{1}{V_m} \sum_{i=1}^{N_c} \mathbf{f}^{*(i)} n_p^{(i)} W(\mathbf{X}^{*(i)}, \mathbf{x}^m), \quad (5.18)$$

where  $W$  is a kernel having compact support that determines the influence of the particle force at a grid node located at  $\mathbf{x}^m$ , and  $V_m$  is the geometric volume of the  $m^{\text{th}}$  grid cell. The reader is referred to Appendix B for complete details on the estimation procedure.

The error involved in the above estimate is composed of forward interpolation error and backward estimation error corresponding to steps 1 and 2 in Section 5.1.3, respectively. The

forward interpolation error is a result of interpolating the fluid velocity that is known at  $M$  nodes to an arbitrary particle location  $\mathbf{X}^{*(i)}$  using Eq. B.2. This interpolated value is denoted  $\{\mathbf{U}^f(\mathbf{X}^{*(i)})\}_M$  (subscript  $M$  represents the number of grid nodes). A global rms forward interpolation error in estimating  $\mathbf{U}_f(\mathbf{X}^{*(i)})$  is defined as

$$\epsilon_U = \sqrt{\frac{\sum_{i=1}^{N_c} (\{\mathbf{U}_f(\mathbf{X}^{*(i)})\}_M - \mathbf{U}_f(\mathbf{X}^{*(i)}))^2}{N_c}} \propto \frac{1}{M^p}, \quad (5.19)$$

which scales as  $M^{-p}$  with grid size, where the exponent  $p$  depends on the order of convergence of the numerical scheme. Although we use data from  $N_c$  particles to compute this error, the forward interpolation error scales purely with grid size (independent of number of particles). The error from forward interpolation is reported in section 5.4.1.1.

In this study we are interested in characterizing the individual contributions to total numerical error in the estimate  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  from forward interpolation (step 1) and backward estimation (step 2). In order to isolate and quantify the backward estimation error incurred by the four different schemes, we need the forward interpolation error to remain the same when forming the estimate  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$ . This is achieved by exploiting the fact that the fluid velocity field is analytically specified by Eq. 5.17 in the entire domain. In the rest of the error analysis that follows for the numerical estimation of  $\mathbf{F}^{fp}(\mathbf{x}, t)$ , it is assumed that the fluid velocity is obtained from the analytical expression and therefore, the error in the estimate arises only from backward estimation. In all the tests that report backward estimation errors (Sec. 5.4.1.2 through end of Sec. 5.4.2), the exact analytical expression for  $\mathbf{U}^f(\mathbf{X}^*)$  is used in Eq. 5.14 to calculate  $\mathbf{f}^*$ .

The numerical estimate  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  is a random variable, and its difference from  $\mathbf{F}^{fp}(\mathbf{x})$  measured in  $p$ -norm defines the total numerical error:

$$\epsilon_F \equiv \left\| \{\mathbf{F}_m^{fp}\}_{N_{pc},M} - \langle \mathbf{F}_m^{fp} \rangle \right\|_p. \quad (5.20)$$

This total numerical error contains contributions from finite grid resolution and finite number of computational particles. Whereas in standard CFD finite-difference/finite-volume codes it is sufficient to reduce the grid size and time step to empirically establish numerical convergence,

this difference in the dependence of the numerical error requires a new approach to establishing numerical convergence of LE calculations.

Many LE numerical studies employ the conventional CFD approach to establish numerical convergence. However, simply increasing the grid resolution by increasing  $M$  while keeping the total number of computational particles  $N_c$  fixed does not result in a monotonic decrease of the total error. This is because as  $M$  is increased for fixed  $N_c$ , the number of computational particles per cell  $N_{pc} = N_c/M$  decreases. Decreasing  $N_{pc}$  means fewer samples per cell, and this results in higher statistical error (which characterizes the level of fluctuations in the random estimate  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$ ) for grid-based estimation methods. On the other hand, while decreasing the total number of cells  $M$  with fixed total number of computational particles does decrease the statistical error, it is at the cost of increasing spatial discretization error.

Most numerical studies seek to establish convergence of LE simulations by increasing the total number of particles  $N_c$ . For a fixed total number of particles there exists an optimal choice of grid size that minimizes the total numerical error. Clearly, a complete characterization of the individual contributions to total error from finite number of particles and finite grid size is essential to determine the optimal choice of numerical parameters for any scheme. This motivates an error decomposition that is described below.

### 5.3.1 Error Decomposition

We decompose the numerical error using an approach similar to that employed by Xu and Pope (1999). For our test problem, only the  $x$ -component of the force contributes to the error in Eq. 5.20 which is decomposed as

$$\epsilon_F \equiv \{F_{x,m}^{fp}\}_{N_{pc},M} - \langle F_{x,m}^{fp} \rangle = \Sigma_F + D_F = \Sigma_F + B_F + S_F, \quad (5.21)$$

where  $\Sigma_F$  is the statistical error, and  $D_F$  is the deterministic error. The deterministic error  $D_F$  is further decomposed into bias  $B_F$  and discretization  $S_F$  error components.

The finite number of particles used in Eq.5.18 to generate a random estimate of the mean interphase momentum transfer term  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  results in statistical fluctuations of the estimate about its expected value. The statistical error  $\Sigma_F$ , arising from these fluctuations, is

defined as

$$\Sigma_F \equiv \{\mathbf{F}_m^{fp}\}_{N_{pc},M} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle. \quad (5.22)$$

The statistical error is assumed to follow a normal distribution, and is modeled as

$$\Sigma_F = \frac{c_F \theta}{\sqrt{N_{pc}}}, \quad (5.23)$$

where  $c_F$  is the statistical error coefficient and  $\theta$  is a standardized normal random variable. Xu and Pope (1999) note in their calculations that the statistical error  $\Sigma_F$  converges as  $N_{pc}^{-1/2}$  and an identical behavior has been seen in other PDF/Monte Carlo simulations (Pope, 1995; Welton and Pope, 1997). Clearly the statistical error decreases as the number of computational particles per cell  $N_{pc}$  increases, and for sufficiently high  $N_{pc}$  we expect  $c_F$  to be a constant independent of  $N_{pc}$ . As we shall see later, statistical error can also be decreased by performing multiple independent simulations (MIS) with the same nominal  $N_{pc}$  per simulation, and then averaging over the MIS.

The bias error  $B_F$  is defined as the deterministic error resulting from finite number of particles and is written as:

$$B_F \equiv \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle - \{\mathbf{F}_m^{fp}\}_{\infty,M}. \quad (5.24)$$

Numerical experiments and analysis (Pope, 1995; Xu and Pope, 1999) validate the following model for the bias error:

$$B_F = \frac{b_F(M)}{N_{pc}}, \quad (5.25)$$

where the bias coefficient  $b_F$  indicates the magnitude of bias for a given  $N_{pc}$ . Note that the bias coefficient  $b_F$  is assumed to be a function of the grid size through  $M$ . It is important to note that MIS can reduce statistical error, but not the bias error. The only way to reduce bias error is to increase  $N_{pc}$ .

Finally, the discretization error is identified as the remaining deterministic error in  $\epsilon_F$ , such that

$$S_F \equiv \{\mathbf{F}_m^{fp}\}_{\infty,M} - \langle \mathbf{F}_m^{fp} \rangle. \quad (5.26)$$

The discretization error for most finite-difference/finite-volume CFD schemes with power-law truncation error dependence can be modeled as

$$S_F = \frac{a_F}{M^p}, \quad (5.27)$$

where  $a_F$  is the discretization error coefficient and the exponent  $p$  depends on the order of convergence of the numerical scheme. In a time-dependent problem, the discretization error will also include a contribution from the temporal discretization error, and all the aforementioned error definitions will be parametrized by the time step  $\Delta t$ , in addition to  $N_{pc}$  and  $M$ .

## 5.4 Results

### 5.4.1 Test 1: Baseline test case

The baseline test case with transcendental mean fluid velocity field, and uniformly distributed particles with delta-function particle velocity distribution was defined in Sec. 5.2. This test is simulated with periodic boundary conditions on the unit cube for each numerical scheme. In all the results presented, the estimate and as well as analytical values for the interphase momentum transfer term are normalized by the maximum analytical value  $\langle \mathbf{F}^{fp} \rangle_{\max}$  in the domain. Therefore, all the individual error contributions are also normalized.

#### 5.4.1.1 Forward Interpolation Error

Figure 5.4 shows the forward interpolation error defined by Eq. 5.19 as a function of cell size  $h = (\Delta x \Delta y)^{1/2}$ . In the results shown, the number of particles per cell  $N_{pc} = 100$ , and the grid varies from  $21 \times 21 \times 4$  nodes to  $61 \times 61 \times 4$  nodes. The figure shows that the fourth-order LPI-4 scheme is the most accurate of all the schemes, and it also has the highest rate of convergence. The least accurate scheme is PCA, and the reason is because this approximation scheme does not exactly recover nodal values, unlike the other three schemes that are true interpolation schemes. The PCA results converge with second-order accuracy, as expected. Since TSE uses linear basis functions for forward interpolation (see Appendix B), its accuracy and convergence are identical to that of LPI-2. The data show that the numerical schemes

follow their theoretical rates of convergence, which is four for LPI-4, and two for LPI-2, TSE, and PCA (Monaghan, 1992).

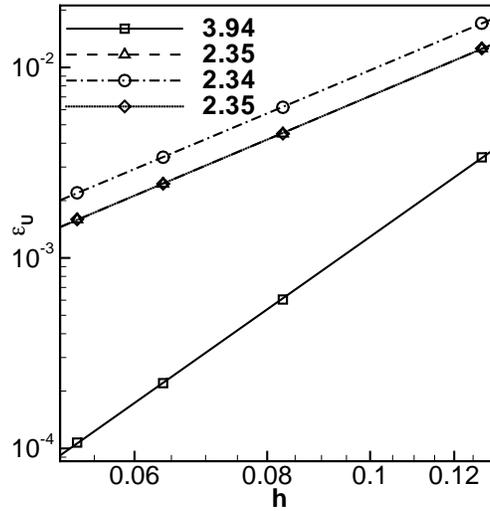


Figure 5.4 Convergence of forward interpolation error  $\epsilon_U$  with grid spacing  $h$ .  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The values in the legend are the slope of linear least-squares fit to the data. These are close to the order of the schemes.

#### 5.4.1.2 Statistical Error

Although Eq. 5.22 provides a formal definition of the statistical error, actually computing the statistical error requires a numerical estimate of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle$ . We estimate  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle$  by performing  $\mathcal{M}$  multiple independent realizations, each with the same  $N_{pc}$  and on the same grid  $M$  but initialized with different random seeds, and taking the arithmetic mean of the MIS. An estimate of the statistical error  $\widehat{\Sigma}_F$  can now be obtained by replacing  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle$  in Eq. 5.22 with its MIS estimate  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}}$ , to get

$$\widehat{\Sigma}_F \equiv \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}}. \quad (5.28)$$

The scaling of this estimate for the statistical error with  $\mathcal{M}$  is revealed by defining  $\Sigma_{F,\mathcal{M}} \equiv \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},\mathcal{M}} \rangle_{\mathcal{M}} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},\mathcal{M}} \rangle$ , and rewriting  $\widehat{\Sigma}_F$  as

$$\widehat{\Sigma}_F = \Sigma_F - \Sigma_{F,\mathcal{M}} = \frac{c_F \theta}{\sqrt{N_{pc}}} - \frac{c_F \xi}{\sqrt{(\mathcal{M}N_{pc})}}, \quad (5.29)$$

where  $\theta$  and  $\xi$  are independent standardized normal random variables. (See Appendix C for details.) The scaling shows that  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},\mathcal{M}} \rangle_{\mathcal{M}} \rightarrow \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},\mathcal{M}} \rangle$  as  $\mathcal{M}^{-1/2}$  for sufficiently large  $\mathcal{M}$ , and this is true for any  $N_{pc} > 1$ . This  $\mathcal{M}^{-1/2}$  scaling is verified by the plot in Fig. 5.5(a) of  $\text{rms}(\widehat{\Sigma}_F)$  as a function of  $\mathcal{M}$  at  $(x = 0.5, y = 0.5)$  for a fixed number of particles  $N_{pc} = 100$  and a  $21 \times 21 \times 4$  grid. The slopes of the least-squares line fits to the data from all schemes are close to  $-0.5$ . This plot also tells us that using MIS we can reduce the statistical error  $\Sigma_{F,\mathcal{M}}$ , which scales as  $(\mathcal{M}N_{pc})^{-1/2}$ , to negligible levels compared to the other error contributions.

We now verify the dependence of  $\Sigma_{F,\mathcal{M}}$  on  $N_{pc}$  that is predicted by Eq. 5.29. Fig. 5.5(b) shows the variation of  $\text{rms}(\widehat{\Sigma}_F)$  with  $N_{pc}$  for a fixed number of realizations  $\mathcal{M} = 100$  on a  $21 \times 21 \times 4$  grid at a representative location  $(x = 0.5, y = 0.5)$ . The slopes of the least-squares line fits to the data are all close to  $-0.5$ , thus verifying the expected convergence of the statistical error as  $N_{pc}^{-1/2}$  that is predicted by Eq. 5.29. For the baseline case of zero particle velocity variance there is little difference in the statistical error incurred by various schemes. Even with just 10 particles per cell the statistical error is  $O(10^{-4})$ .

When computing the contributions from bias error and deterministic error to the total error using Eq. 5.21, the statistical error will need to be negligibly small in comparison. From Eq. 5.28 we can infer that if the product  $N_{pc}\mathcal{M}$  is sufficiently large, then the statistical error can be made arbitrarily small. In this case the estimate for average interphase momentum transfer term  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},\mathcal{M}} \rangle_{\mathcal{M}}$  will have a very small and spatially uniform statistical error for different values of  $N_{pc}$ . The results for deterministic and bias error presented in the following subsections correspond to  $N_{pc}\mathcal{M} = 60,000$ , which ensures very low statistical error.

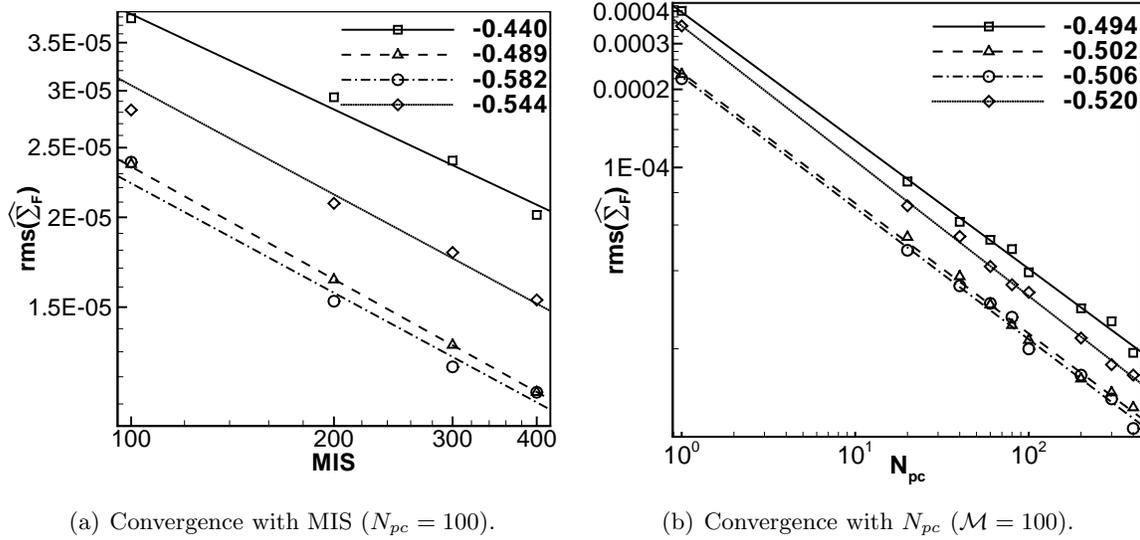


Figure 5.5 Convergence of statistical error  $\text{rms}(\hat{\Sigma}_F)$  with (a) number of multiple independent simulations  $\mathcal{M}$ , and (b) number of particles per cell  $N_{pc}$ , at  $(x = 0.5, y = 0.5)$  for Test 1 with zero particle velocity variance ( $V_1^* \stackrel{D}{=} \mathcal{N}[1.0, 0.0]$ ).  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The values in the legend are the slope of linear least-squares fit to the data.

### 5.4.1.3 Deterministic Error

The total deterministic error  $D_F$  is estimated by  $\hat{D}_F$  using an ensemble-averaged estimator at finite  $N_{pc}$  and the analytical solution:

$$\hat{D}_F = \left| \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, \mathcal{M}} \rangle_{\mathcal{M}} - \langle \mathbf{F}_m^{fp} \rangle \right|. \quad (5.30)$$

The contour plot of deterministic error estimated by Eq. 5.30 is shown in Fig. 5.6 for a  $21 \times 21 \times 4$  grid with  $N_{pc} = 400$  using  $\mathcal{M} = 150$  independent realizations ( $N_{pc}\mathcal{M} = 60,000$ ). Of the four

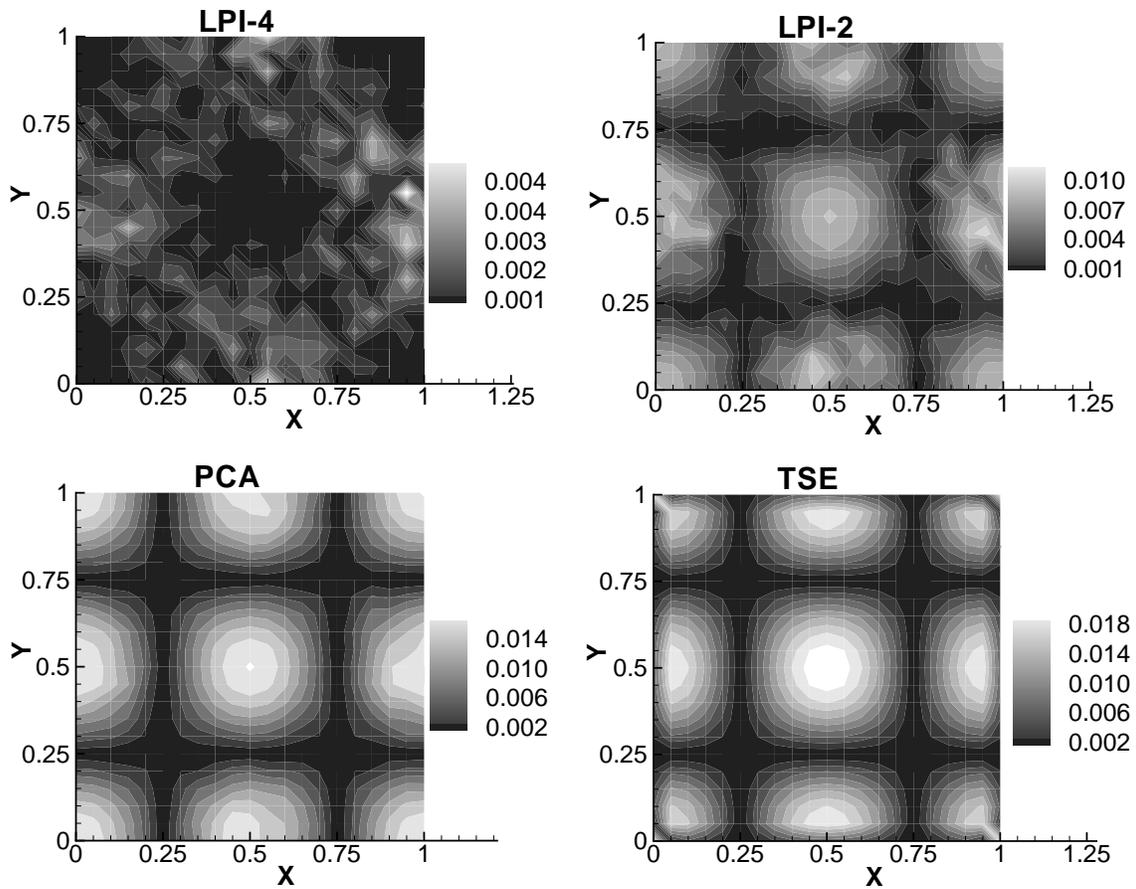


Figure 5.6 Contour plot of total deterministic error for Test 1 on a  $21 \times 21 \times 4$  grid with  $N_{pc} = 400$  and  $N_{pc}\mathcal{M} = 60,000$ . The fourth-order LPI-4 scheme exhibits least error. All schemes show considerable spatial variation with an order of magnitude difference in the total deterministic error across the domain.

schemes, LPI-4 incurs the least total deterministic error and the magnitude of error incurred by the other three schemes is comparable. The figure also shows that the location of the maximum deterministic error is not the same for all schemes. Therefore, a comparison of deterministic error incurred by the different schemes at a fixed location can be misleading.

#### 5.4.1.4 Bias Error

Bias error, which is defined by Eq. 5.24 in Section 5.3.1, is that part of the deterministic error resulting from a finite number of particles. Similar to the estimation of expected values that arise in the statistical error, the quantity  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle$  in the bias error is approximated by an ensemble-averaged estimate  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}$ . Therefore, the approximate expression for bias error  $\hat{B}_F$  is

$$\hat{B}_F = \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}} - \{\mathbf{F}_m^{fp}\}_{\infty,M} = \frac{b_F(M)}{N_{pc}}. \quad (5.31)$$

In order to calculate the bias error based on this definition, it is necessary to compute  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  at  $N_{pc} = \infty$  which is impractical and computationally prohibitive. However, noting that the magnitude of the bias coefficient  $b_F$  is a function of only the grid size  $M$ , we can use two evaluations of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}$  at  $N_{pc} = N_{pc}^{(1)}$  and  $N_{pc}^{(2)}$  to calculate  $b_F$  as follows:

$$b_F(M) = \frac{N_{pc}^{(1)} N_{pc}^{(2)}}{N_{pc}^{(2)} - N_{pc}^{(1)}} \left( \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}^{(1)},M} \rangle_{\mathcal{M}} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}^{(2)},M} \rangle_{\mathcal{M}} \right). \quad (5.32)$$

If more than two values of  $N_{pc}$  are used, the slope obtained from a linear least squares fit to the data yields the bias coefficient  $b_F$ .

Although there is considerable spatial variation of the bias error, the variation of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}$  with  $N_{pc}^{-1}$  is shown in Fig. 5.7 at the same representative location ( $x = 0.5, y = 0.5$ ) where the statistical error scaling was shown. From the figure, the linear behavior of bias with  $N_{pc}^{-1}$  is apparent. Since the total deterministic error exhibits different spatial distribution for each scheme, a contour plot of the bias coefficient  $b_F$  is more informative. The bias coefficient is calculated using Eq. 5.32, and Fig. 5.8 shows that TSE is the least biased estimator (by two orders of magnitude compared to the other schemes considered) followed by PCA, LPI-2, and LPI-4.

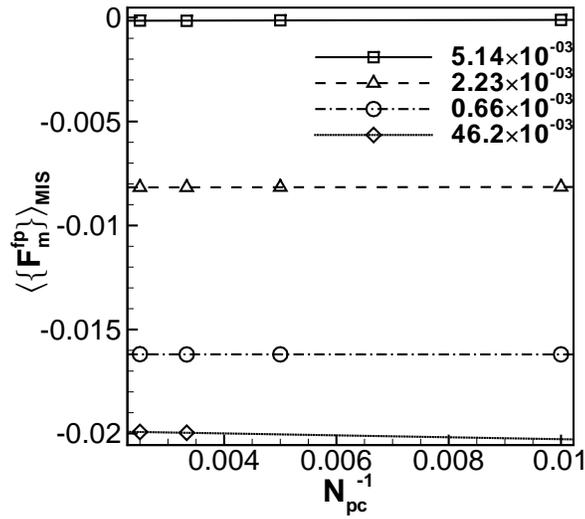


Figure 5.7 Estimation of bias coefficient  $b_F$  from plot of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}$  as a function of  $N_{pc}^{-1}$  for  $N_{pc}\mathcal{M} = 60,000$  at  $(x = 0.5, y = 0, 5)$ .  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit which is also equal to bias coefficient is indicated in the legend.

#### 5.4.1.5 Discretization error

Discretization error defined by Eq. 5.26 depends only on the spatial resolution, or grid size,  $h$ . A smaller value of  $h$  (more grid points) for a given  $N_{pc}$  will yield a more resolved mean field, and hence a lower discretization error.

Similar to observations made for bias error, if the discretization error is estimated based on its definition (cf. Eq. 5.26), then one needs to calculate  $\{\mathbf{F}_m^{fp}\}_{\infty,M}$  which is impractical. Therefore, we calculate  $S_F$  by forming an approximate estimate for  $\{\mathbf{F}_m^{fp}\}_{\infty,M}$  denoted by  $\{\mathbf{F}_m^{fp}\}_e$ . For a fixed grid size  $M$  with known bias coefficient  $b_f$  (see Eq. 5.31), the estimate of  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  in the limit of  $N_{pc}$  going to infinity is obtained by Richardson extrapolation (Xu and Pope, 1999) of  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}$  at two or more values of  $N_{pc}$ . The expression for  $\{\mathbf{F}_m^{fp}\}_e$  is

$$\{\mathbf{F}_m^{fp}\}_e = \frac{\sum_{i=1}^{\mathcal{N}} \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}^{(i)},M} \rangle_{\mathcal{M}} - b_F \sum_{i=1}^{\mathcal{N}} (N_{pc}^{(i)})^{-1}}{\mathcal{N}} \cong \{\mathbf{F}_m^{fp}\}_{\infty,M}, \quad (5.33)$$

where  $\mathcal{N}$  is the number of  $N_{pc}$  values for which the ensemble-averaged estimates are formed

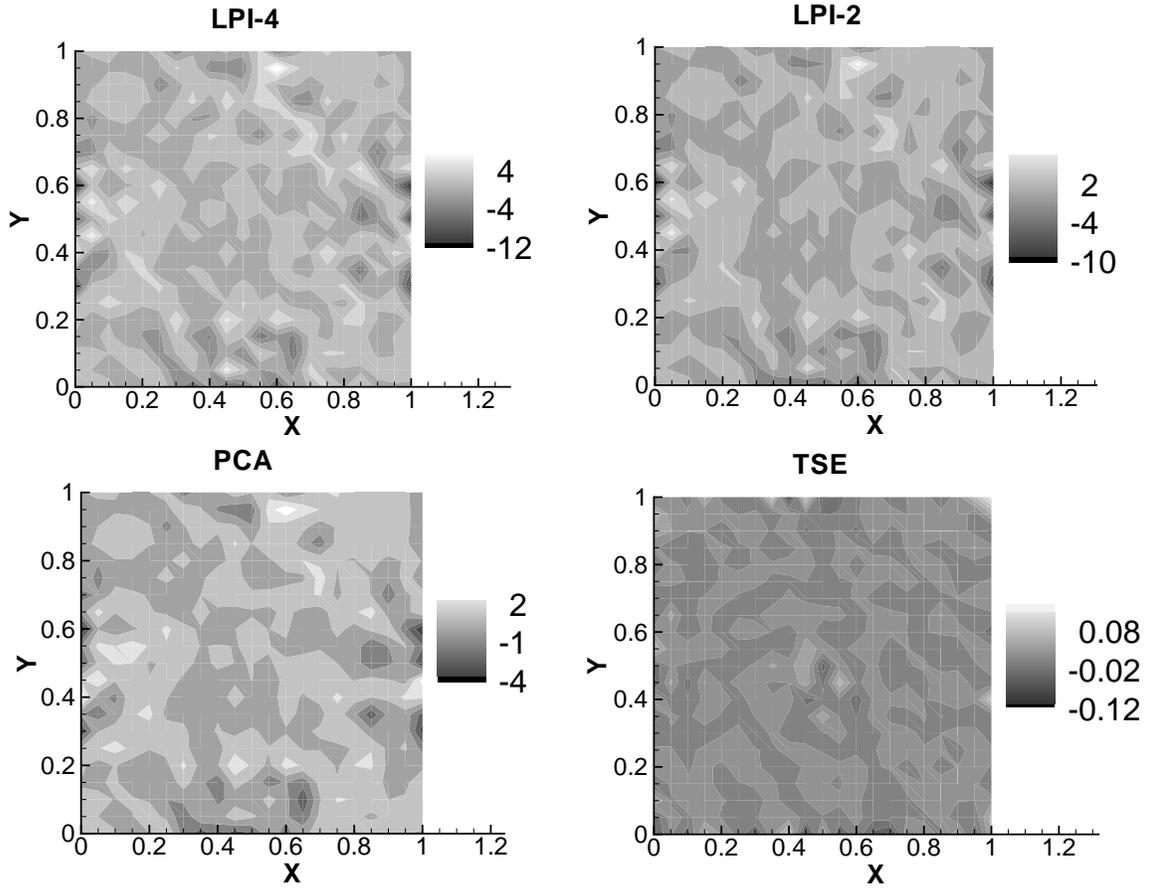


Figure 5.8 Contour plot of the bias coefficient  $b_F$  for Test 1 on a  $21 \times 21 \times 4$  grid ( $N_{pc}\mathcal{M} = 60,000$ ). TSE incurs the least bias error with a bias coefficient that is two orders of magnitude lower than the other schemes.

for each grid size  $M$ . In the estimation of bias coefficient, the effect of statistical fluctuations is minimized by choosing  $N_{pc}\mathcal{M} = 60,000$ . Furthermore, in the above expression for estimating  $\{\mathbf{F}_m^{fp}\}_e$ , the effect of bias is also removed from the numerical estimate. The effects of both statistical fluctuations and bias error are minimized in the estimate of  $\{\mathbf{F}_m^{fp}\}_e$ ; thus,  $\{\mathbf{F}_m^{fp}\}_e$  is a good approximate estimate for  $\{\mathbf{F}_m^{fp}\}_{\infty, M}$ .

Figures 5.9(a) and 5.9(b) show the convergence of spatial discretization error  $S_F = \left| \{\mathbf{F}_m^{fp}\}_e - \langle \mathbf{F}_m^{fp} \rangle \right|$  with cell size  $h = (\Delta x \Delta y)^{1/2}$  at two representative  $(x, y)$  locations (0.5, 0.5) and (0.6, 0.2), respectively. Figure 5.9(a) shows that at the (0.5, 0.5) location, LPI-4 incurs

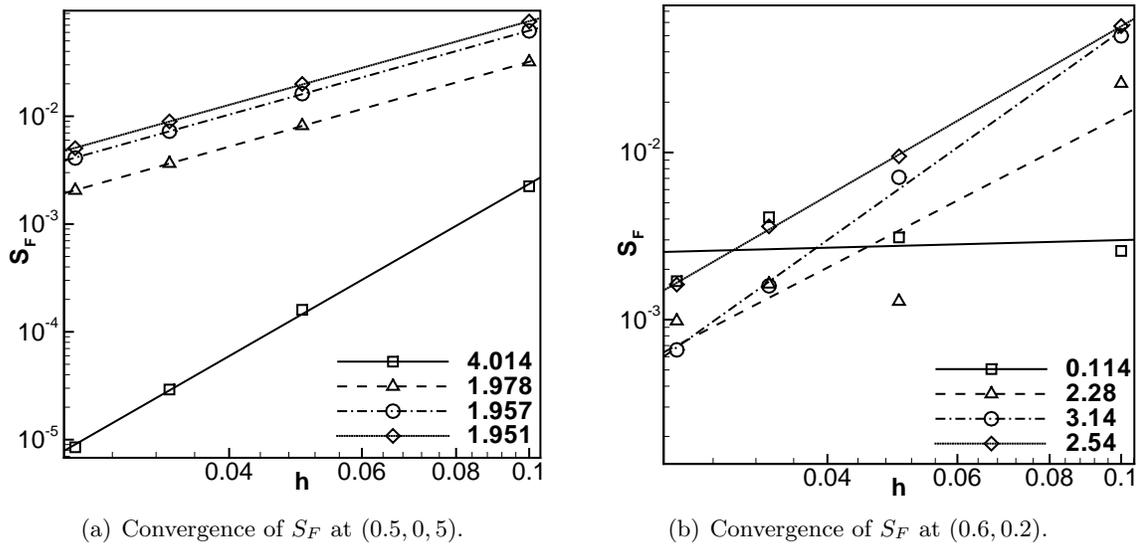


Figure 5.9 Convergence of spatial discretization error  $S_F = \left| \{\mathbf{F}_m^{fp}\}_e - \langle \mathbf{F}_m^{fp} \rangle \right|$  with grid spacing  $h$  at different spatial locations to illustrate the strong spatial nonuniformity in convergence characteristics of the schemes: (a) convergence of  $S_F$  with  $h$  at  $(x = 0.5, y = 0.5)$  for Test 1, (b) convergence of  $S_F$  with  $h$  at  $(x = 0.6, y = 0.2)$  for Test 1.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend.

the least discretization error and has the highest rate of convergence; followed by LPI-2, PCA, and TSE. The convergence rates are once again very close to the theoretical values and show trends similar to those observed for forward interpolation error in Sec. 5.4.1.1. At the (0.6, 0.2) location, however, Fig. 5.9(b) shows that the convergence rate of all schemes have changed considerably. LPI-4 now has the lowest rate of convergence. PCA, on the other hand, is the fastest converging scheme followed by TSE and LPI-2.

Since the rate of convergence of spatial discretization error (Fig. 5.9) exhibits strong spatial nonuniformity for each scheme, a global discretization error

$$\hat{S}_F = \frac{1}{M} \sqrt{\sum_{m=1}^M (\{\mathbf{F}_m^{fp}\}_e - \langle \mathbf{F}_m^{fp} \rangle)^2}, \quad (5.34)$$

is defined using the standard rms technique given by Eq. 5.34, with  $M = M_x \times M_y \times M_z$  being the total number of grid nodes. Figure 5.10 shows the convergence of global discretization

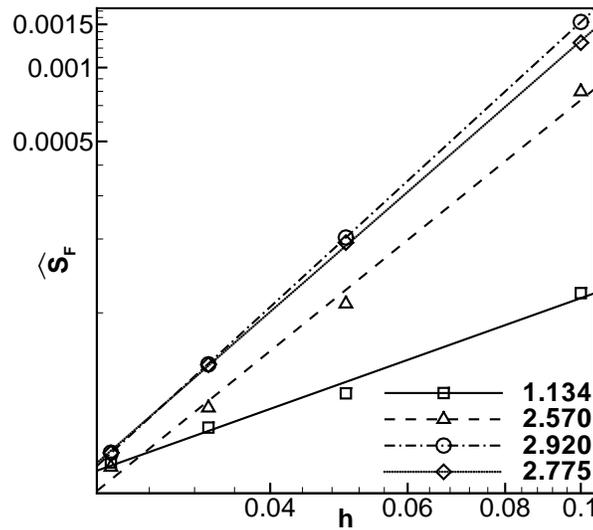


Figure 5.10 Convergence of global discretization error  $\hat{S}_F$  with grid spacing  $h$  for Test 1.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend. Note the dramatic difference in convergence rates that deviate considerably from the order of the schemes, in contrast to Fig. 5.4.

error  $\hat{S}_F$  with grid spacing  $h$ . The figure reveals a deficiency in LPI-4, which is no longer

the fastest converging scheme. This sharp fall in the convergence rate of global discretization error incurred by LPI-4 can be explained as a result of strong spatial nonuniformity of local convergence rates observed in Fig. 5.9. On a coarse grid (high value of  $h$ ), LPI-4 is the most accurate estimator but it is least accurate on fine grids (at lower  $h$ ) due to the slow rate of convergence of its global discretization error. PCA exhibits the highest rate of convergence of global discretization error, followed by TSE, LPI-2, and LPI-4.

Our results for this test case of zero particle velocity variance show that the statistical error for all schemes is of the same order of magnitude. However, based on the magnitude of bias error, and the rate of convergence of global discretization error, TSE and PCA stand out as the preferred methods for estimating the mean interphase momentum transfer term.

#### 5.4.2 Test 2: Effect of nonzero particle velocity variance

In this test we investigate the effect of nonzero particle velocity variance that is representative of many practical particle-laden flows. Nonzero particle velocity variance may arise as a result of turbulence. The only change from the baseline test is the particle velocity distribution, which is now specified to be a normal with nonzero variance:

$$V_1^* \stackrel{D}{=} \mathcal{N}[\langle V_1 \rangle, \sigma_{V_1}^2] \equiv [1.0, 0.3]. \quad (5.35)$$

Since the particle velocity distribution now has a finite variance, it is expected that the statistical error in estimating the mean interphase momentum transfer term will be larger than that incurred in the baseline test case with zero variance (cf. Fig 5.5).

Figure 5.11 shows the statistical error incurred by the various schemes for finite particle velocity variance, but with all other test conditions identical to those in Fig. 5.5. As expected, the statistical error shows an increase for all the schemes (by at least one order of magnitude), but the increase is not the same for all schemes. While in the baseline test with zero particle velocity variance the statistical error is  $O(10^{-4})$  even with just 10 particles per cell for all the schemes (cf. Fig. 5.5(b)), Fig. 5.11(b) shows that for  $\sigma_{V_1}^2 = 0.3$  with 10 particles per cell the statistical error is at least an order of magnitude higher  $O(10^{-3} - 10^{-2})$ . If the particle velocity variance is not zero, Fig. 5.11(b) also shows that the LPI-4 and LPI-2 schemes incur statistical

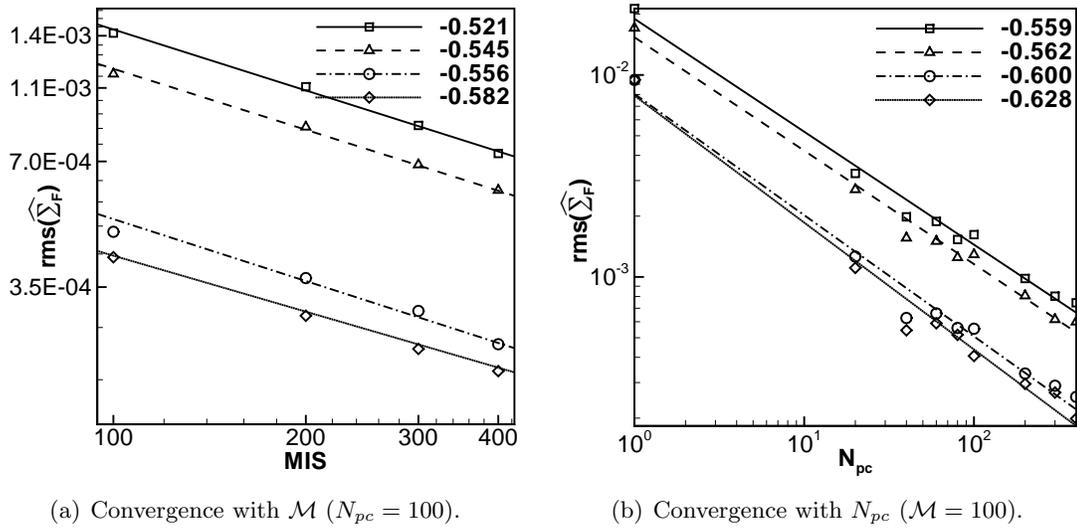


Figure 5.11 Convergence of statistical error  $\text{rms}(\hat{\Sigma}_F)$  with (a) number of multiple independent simulations  $\mathcal{M}$ , and (b) number of particles per cell  $N_{pc}$ , at  $(x = 0.5, y = 0.5)$  for Test 2 with nonzero particle velocity variance ( $V_1^* \stackrel{D}{=} \mathcal{N}[1.0, 0.3]$ ).  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least squares fit is indicated in the legend.

error that is an order of magnitude larger than that incurred by TSE or PCA. The difference in the statistical error incurred by the schemes persists even with increasing MIS, as shown in Fig. 5.11(a).

The bias coefficient  $b_F$  contours for Test 2 exhibit similar spatial variation for all the schemes as seen in Test 1 (see Fig. 5.8), and are hence not presented here. However, an important effect of nonzero particle velocity variance is to significantly increase the bias coefficient values for TSE (min/max:  $-2/0.5$ ), whereas those of LPI-4 (min/max:  $-10/6$ ), LPI-2 (min/max:  $-8/4$ ), and PCA (min/max:  $-3/3$ ) remain at levels similar to that seen in Test 1. Nevertheless, even for this test TSE incurs the least bias error of all the schemes considered.

Fig. 5.12 shows the variation of global discretization error  $\hat{S}_F$  with  $h$ , which when compared to the previous case of zero variance (Fig. 5.10) shows that the magnitude of global discretization error and as well as its rate of convergence remain nearly the same. This observation is not surprising because the only difference between the two test cases is in the particle velocity distribution, which should not affect the discretization error.

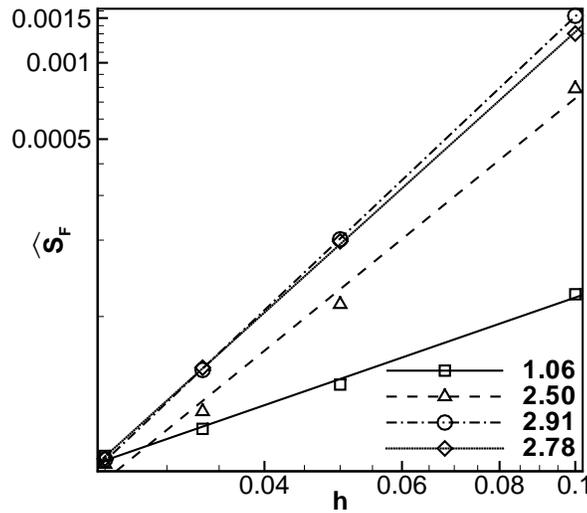


Figure 5.12 Convergence of global discretization error  $\hat{S}_F$  with grid spacing  $h$  for Test 2.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE. The slope of the linear least-squares fit is indicated in the legend. Comparison with Fig. 5.10 reveals that the effect of nonzero particle velocity variance on the magnitude and convergence rate of the global discretization error is minimal.

Both TSE and PCA outperform LPI-4 and LPI-2 in terms of statistical error and incur relatively low bias error. Also, the rate of convergence of global discretization error for TSE and PCA is nearly twice that of LPI-4. The results of this test reaffirm the conclusions of Test 1 that both TSE and PCA are the best schemes to estimate the mean interphase momentum transfer term.

#### 5.4.3 Test 3: Variation of particle position distribution

In our numerical tests thus far we have chosen the particles to be uniformly distributed in physical space. However, in realistic particle-laden flows the particles will not be uniformly distributed in general. In this test, the fluid velocity field and particle velocity distribution are retained as in test 2 (finite variance). In order to ascertain the effect of non-uniform particle

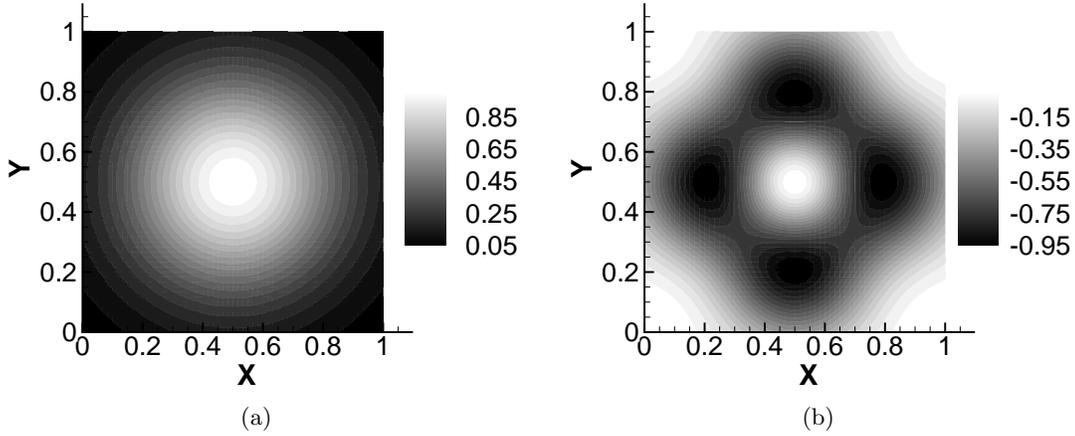


Figure 5.13 Test 3 with nonuniform particle position distribution: (a) Contour plot of the scaled number density  $n(x, y)/n_o$ . (b) Contour plot of the scaled analytical mean interphase momentum transfer term  $\langle F_x^{fp} \rangle / \langle F_x^{fp} \rangle_{\max}$ .

distribution, the particle number density field is specified to be

$$n(x, y) = n_o \exp \left( - \frac{(x - \mathcal{L}_x/2)^2 + (y - \mathcal{L}_y/2)^2}{\mathcal{L}_x^2/16 + \mathcal{L}_y^2/16} \right), \quad (5.36)$$

where  $n_o$  is a constant so chosen such that there are a finite number of particles near the boundary cells. Fig. 5.13(a) shows the contour plot for  $n(x, y)/n_o$ . Using the particle position pdf  $f_{\mathbf{x}} = n(\mathbf{x}) / \langle N_p \rangle$  implied by the number density in Eq. 5.36, the analytical expression for normalized mean interphase momentum transfer term is obtained from Eq. 5.16. Fig. 5.13(b) shows the resulting normalized mean interphase momentum transfer term. In this inhomogeneous test case no attempt is made to decompose the various numerical errors, but only the error resulting from the averaged estimate obtained from multiple realizations along with the 95% confidence intervals are presented. The 95% confidence interval (Xu and Pope, 1999) for the estimation of interphase momentum transfer term is estimated as

$$\delta = 1.69 \left[ \frac{1}{\mathcal{M} - 1} \sum_{i=1}^{\mathcal{M}} \left( \{ \mathbf{F}_m^{fp} \}_{N_{pc}, M}^{(i)} - \langle \{ \mathbf{F}_m^{fp} \}_{N_{pc}, M} \rangle_{\mathcal{M}} \right)^2 \right]^{1/2}. \quad (5.37)$$

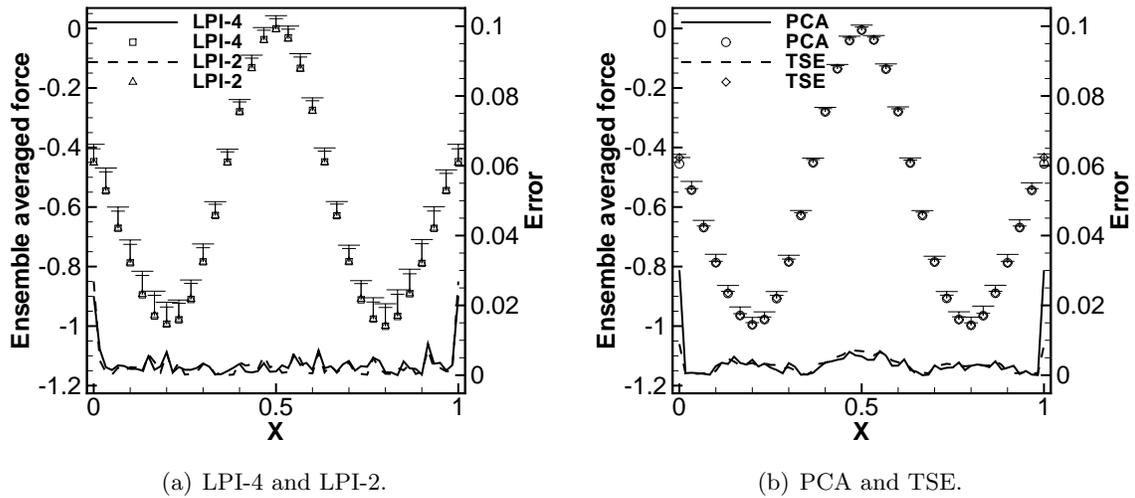


Figure 5.14 Ensemble-averaged mean interphase momentum transfer term for Test 3 with confidence intervals, and its error with respect to analytical value. In both panels the symbols indicate  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}}$  (scale on left vertical axis) as a function of  $x$  at  $y = z = 0.5$ . Error bars denote 95% confidence intervals above and below the mean value, but are shown only above for clarity. The error bars for the two schemes in each panel are distinguished by the length of the cross-bars, with (a) LPI-4 (long), LPI-2 (short), and (b) PCA (long), TSE (short). Lines in both panels indicate the error  $|\langle \mathbf{F}_m^{fp} \rangle - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}}|$  whose scale is given on the right vertical axis.

The test is carried out on a  $61 \times 61 \times 4$  grid and 200 independent but identical simulations are performed in order to calculate the 95% confidence intervals. In Fig. 5.14(a), the symbols indicate the ensemble-averaged mean interphase momentum transfer term  $\langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}}$  obtained using LPI-4 and LPI-2. The scale for the symbols is on the left vertical axis. The height of the error bars indicate the 95% confidence intervals on the ensemble average, and in order to distinguish between the two cases the error bars with the longer cross-bar indicate LPI-4. In the same figure, the lines represent  $\left| \langle \mathbf{F}_m^{fp} \rangle - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc}, M} \rangle_{\mathcal{M}} \right|$ , the error between the ensemble-averaged and analytical mean interphase momentum transfer term. The scale for the error is on the right vertical axis. Since the number density variation in  $x$  and  $y$  is identical, the ensemble-averaged mean interphase momentum transfer term and error are reported along  $x$ , for  $y = z = 0.5$ . Figure 5.14(b) shows results for the same test as Fig. 5.14(a), but for the PCA and TSE schemes. The longer cross-bars on the 95% confidence intervals correspond to PCA.

For all the schemes considered, the size of the confidence interval shows an increase in the regions of low number density. This increase is maximum for LPI-4 followed by LPI-2, PCA, and TSE. The error shows the same trend in the regions of low number density. This test shows the advantage of using TSE and PCA over LPI-4 and LPI-2 for particle-laden systems with non-uniform number density.

#### 5.4.4 Test 4: Variation of fluid velocity field

Depending on the type of simulation (CFD, LES or DNS), the spectral content of the fluid velocity field will be different. In this test, the fluid velocity field is chosen to be a sinusoidal field given by Eq. 5.38 and its wavelength is varied by increasing  $f_o$  from a minimum value of 1 to a maximum value of 25. This test reveals the accuracy of mean interphase momentum transfer term estimation with variation in the fluid velocity spectrum

$$U_1^f(x, y) = \cos\left(\frac{2\pi f_o x}{\mathcal{L}_x}\right) \cos\left(\frac{2\pi f_o y}{\mathcal{L}_y}\right). \quad (5.38)$$

The ensemble-averaged summed mean square of the total error, denoted by  $\hat{\epsilon}_F$ , and given

by the following expression

$$\hat{\epsilon}_F = \frac{1}{\mathcal{M}} \sum_{i=1}^{\mathcal{M}} \frac{\sum_{m=1}^{M^3} \left( \{\mathbf{F}_m^{fp}\}^i - \langle \mathbf{F}_m^{fp} \rangle \right)^2}{M^3} \quad (5.39)$$

is calculated for different values of  $f_o$ .

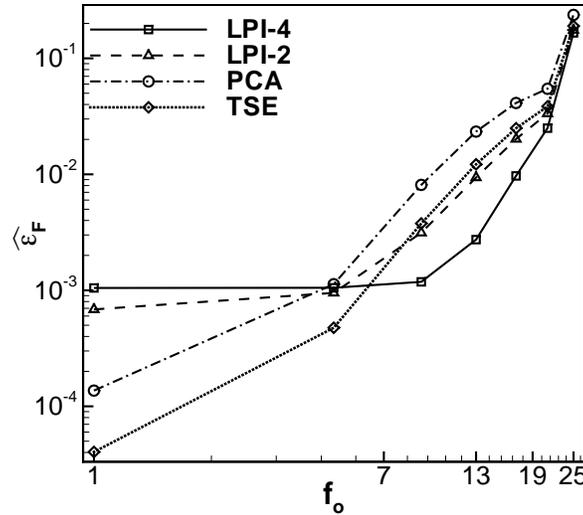


Figure 5.15 Ensemble-averaged summed mean square error  $\hat{\epsilon}_F$  as a function of  $f_o$ , the frequency of the transcendental velocity field.

Particles are uniformly distributed in physical space and the particle velocity field of Test 2 (finite variance) is retained here. The number of computational particles per cell  $N_{pc}$  is 100 and number of realizations  $\mathcal{M} = 150$ . The test is performed on a  $51 \times 51 \times 4$  grid.

Fig. 5.15 shows the variation of  $\hat{\epsilon}_F$  versus  $f_o$  and it can be observed that at low wave numbers TSE and PCA outperform LPI-4 and LPI-2 by an order of magnitude, at intermediate wave numbers LPI-4 and LPI-2 become more accurate than PCA and TSE, and at the highest wavenumber, all the schemes are rather inaccurate and have approximately the same error magnitude. The reduction in accuracy for PCA and TSE from low to intermediate wavenumbers compared to LPI-4 and LPI-2 is attributed to the smoothing property of cubic splines and linear least-squares, respectively. Therefore, for LES and DNS simulations, PCA and TSE will require higher grid resolution to yield the same accuracy as LPI-4 or LPI-2.

## 5.5 Comparison with representative LE numerical parameters

In this study we have performed calculations with very high numerical resolution. The number of particles per cell in our tests typically ranges from 100 to 400. In addition, the number of independent realizations simulated for each test also varies from 100 to 400. However, in most particle-laden simulations of two-way coupled dispersed two-phase flows using LE approach, the number of particles per cell is usually one or even two orders of magnitude lower than the values we have used for our tests, and typically only one realization is simulated. Typical values for the nominal number of particles per cell <sup>4</sup> in 3-D LE simulations range from 0.0156 to 0.125 in Sundaram and Collins (1999) to exactly 1 in Boivin et al. (1998). In 2-D calculations higher  $N_{pc}^n$  values have been used: 3 to 30 in Narayanan et al. (2002) and 16 to 500 in Lakehal and Narayanan (2003). In all but one of these studies (Lakehal and Narayanan, 2003), only one realization is simulated. The contour plot of absolute total error  $\epsilon_F = \left| \{F_{x,m}^{fp}\}_{N_{pc},M} - \langle F_{x,m}^{fp} \rangle \right|$  incurred in the calculation of mean interphase momentum transfer term for Test 2 on a  $61 \times 61 \times 4$  grid with  $N_{pc} = 5$  and  $\mathcal{M} = 1$  is shown by Fig. 5.16. The figure clearly reveals that while all the schemes are unable to provide estimates within 10% error using 5 particles per cell with only one realization, TSE comes closest with a maximum error of only 20%. LPI-4 gives errors as high as 80%. As noted in the introduction, these large numerical errors directly impact the physical insight that is provided by LE simulations. However, it is important to bear in mind that there are other numerical approximations in the fluid flow solver (artificial viscosity and pressure correction) that can mitigate these errors.

## 5.6 Conclusions

Comprehensive testing of four numerical schemes used to estimate mean interphase transfer terms in LE simulations for a novel test problem that admits an analytical solution enables characterization of numerical convergence, as well as accuracy. For estimation of the mean interphase momentum transfer term, all our four tests suggest the use of TSE and PCA, or

<sup>4</sup>The nominal number of particles per cell  $N_{pc}^n$  is defined as the ratio of total number of particles to total number of grid cells

even LPI-2, over LPI-4. TSE and PCA consistently give low statistical and bias errors and yield good estimates even in the regions of low number density. The only exception is when there is high spectral content in the fluid velocity field, where due to the smoothing nature of TSE and PCA, their estimates are less accurate than LPI-4 and LPI-2.

Testing with representative values for the numerical parameters found in typical LE simulations reveals that LPI-4 and LPI-2 incur unacceptably high error, whereas TSE yields the most accurate estimate of all the schemes. The tests validate an error model (cf. Eq. 5.21) of the form

$$\epsilon_F = \frac{c_F \theta}{\sqrt{N_{pc}}} + \frac{b_F(M)}{N_{pc}} + \frac{a_F}{M^p}$$

that decomposes the error into statistical, bias and discretization components, and explicitly characterizes the error in terms of numerical parameters (grid size  $M$  and number of particles per cell  $N_{pc}$ ). An approach to quantifying the values of the coefficients  $a_F$ ,  $b_F$ , and  $c_F$  in the error model is demonstrated for the test problem. If efficient ways to quantify these coefficients are developed for general LE problems, then estimates for the numerical error can be obtained from this model. This can provide the required values of numerical parameters for a given error tolerance.

Our study reveals the need to carefully choose the appropriate numerical scheme for forward interpolation and backward estimation. Although LPI-4 is found to be the preferred scheme for forward interpolation, it results in relatively poor estimates for the mean interphase momentum transfer term. None of the schemes considered is optimal for both forward interpolation and backward estimation. This conclusion also implies that, for the schemes considered, the PIS requirement of Sundaram and Collins (1996)—which states that in order to ensure overall energy balance, the order of the interpolation scheme used in the forward interpolation should be the same as that used in the backward estimation—is at odds with numerical accuracy and convergence requirements.

Spatially nonuniform particle distribution results in poor estimates of mean interphase momentum transfer term in regions where there are fewer particles. If the number density of physical particles becomes zero locally, then TSE will encounter difficulties due to ill-

conditioned matrices. One way to obtain good estimates even in regions of low physical particle number density is by introducing more computational particles in that region. In other words, if the number density of computational particles is maintained relatively uniform during the entire course of simulation, then the statistical error remains uniformly low over the entire domain. This needs to be addressed by computational particle number density control.

In order to obtain numerically converged results, it is imperative to simultaneously reduce the statistical and deterministic error components that result from backward estimation. The bias and statistical error components depend on the number of particles per cell. Therefore, numerical convergence cannot be achieved by grid refinement with a fixed total number of computational particles because the number of particles per cell keeps decreasing. This is because the bandwidth of most numerical schemes scales with the grid spacing<sup>5</sup>. Therefore, it is necessary to keep  $N_{pc}$  fixed in grid resolution studies of LE simulations so that statistical and bias error remain at the same level. While statistical error can be effectively reduced by multiple independent simulations, the same is not true for bias error, which scales as  $N_{pc}^{-1}$ . Time-evolving tests could show higher bias error due to feedback in the particle evolution equations. The only way to reduce bias error is to increase the number of computational particles per cell.

Another important consideration when choosing a numerical scheme for LE calculations is the computational cost involved, and the estimation of additional quantities that may be required. LPI-2 being a second order scheme is the least expensive. PCA and LPI-4 involve the same number of operations while TSE is the computationally most expensive estimation scheme. However, if in addition to the mean interphase momentum transfer term, the estimation of its gradient is also required, then TSE becomes the favored scheme. This is because in TSE, once the interphase momentum transfer term has been calculated, no additional operations are required to compute the gradients. For LPI-4, LPI-2, and PCA, the estimation of gradient amounts to approximately doubling the computational cost that is required for calculating the mean interphase momentum transfer term.

---

<sup>5</sup>An exception is TSE, which is a truly grid-free estimation method.

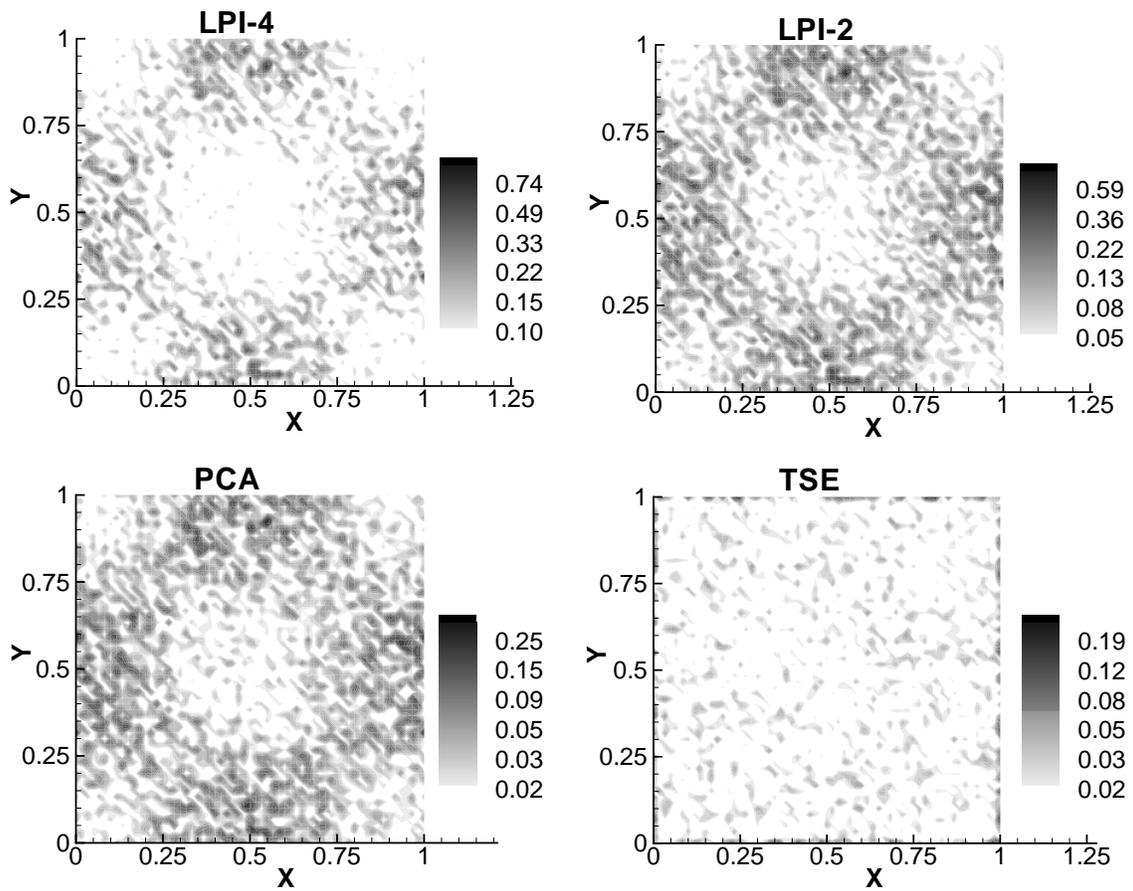


Figure 5.16 Contour plot of absolute total error  $\epsilon_F = \left| \{F_{x,m}^{fp}\}_{N_{pc},M} - \langle F_{x,m}^{fp} \rangle \right|$  for Test 2 with representative LE numerical parameters—a single realization  $M = 1$  with  $N_{pc} = 5$  particles per cell on a  $61 \times 61 \times 4$  grid—to demonstrate the unacceptably large errors incurred by schemes such as LPI-4 and LPI-2.

## CHAPTER 6. A NUMERICALLY CONVERGENT LAGRANGIAN–EULERIAN SIMULATION METHOD FOR DISPERSED TWO–PHASE FLOWS

This chapter is a printed manuscript (Garg et al., 2009b) titled “A numerically convergent Lagrangian–Eulerian simulation method for dispersed two–phase flows” in “Intl. J. Multiphase Flow” authored by R. Garg, C. Narayanan, and S. Subramaniam.

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 (Sundaram and Collins, 1996; Boivin et al., 1998; Narayanan et al., 2002; Patankar and Joseph, 2001; Snider et al., 1998). 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.

## 6.1 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 <sup>1</sup> 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 <sup>2</sup> 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.,

<sup>1</sup>By particle we mean any dispersed-phase element, including solid particles, droplets and bubbles.

<sup>2</sup>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.

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 (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

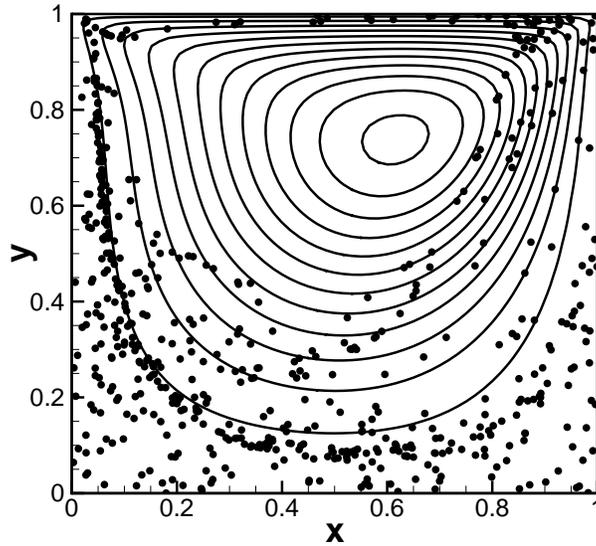


Figure 6.1 Snapshot of a one-way coupled lid-driven cavity flow simulation at non-dimensional time  $tU/L$  equal to 10. Details are provided in Sec. 6.5. The important flow parameters are  $Re = UL/\nu = 100$ ,  $St = \tau_p/\tau_F = 0.8$ . The solid lines represent the fluid phase stream function contours and black dots represent the dispersed-phase particles.

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. Figure 6.1 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.

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; Jaber 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. 6.1, 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*.

## 6.2 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  $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t) \mid i = 1, \dots, N_p(t)\}$ , where  $\mathbf{X}^{(i)}(t)$  denotes the  $i^{th}$  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

$$\frac{d\mathbf{X}^{(i)}}{dt} = \mathbf{V}^{(i)}, \quad (6.1)$$

$$\frac{d\mathbf{V}^{(i)}}{dt} = \frac{\mathbf{f}_p^{(i)}}{m_p^{(i)}} = \mathbf{A}^{(i)}, \quad , i = 1, \dots, N_p(t), \quad (6.2)$$

where  $\mathbf{f}_p^{(i)}$  and  $\mathbf{A}^{(i)}$  are, respectively, the instantaneous force and acceleration experienced by the  $i^{th}$  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

$$f(\mathbf{x}, \mathbf{v}, t) \equiv \langle f'(\mathbf{x}, \mathbf{v}, t) \rangle = \left\langle \sum_{i=1}^{N_p(t)} f'_i(\mathbf{x}, \mathbf{v}, t) \right\rangle = \left\langle \sum_{i=1}^{N_p(t)} \delta_{\mathbf{X}^{(i)}} \delta_{\mathbf{V}^{(i)}} \right\rangle, \quad (6.3)$$

where  $f'$  is the fine-grained density function,  $f'_i$  is the fine-grained density function for the  $i^{\text{th}}$  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. 6.1 and 6.2 implies an evolution equation for  $f(\mathbf{x}, \mathbf{v}, t)$  (Subramaniam, 2001), which is

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x_k} [v_k f] + \frac{\partial}{\partial v_k} [\langle A_k | \mathbf{x}, \mathbf{v}; t \rangle f] = 0. \quad (6.4)$$

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

$$\langle A_k | \mathbf{x}, \mathbf{v}; t \rangle = \frac{1}{f(\mathbf{x}, \mathbf{v}, t)} \left\langle \left\langle \sum_{i=1}^{N_p(t)} A_k^{(i)} f'_i(\mathbf{x}, \mathbf{v}, t) \right\rangle \right\rangle, \quad \text{if } f > 0, \quad (6.5)$$

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

$$\mathbf{A}^{*(i)}(t) = \mathbf{A} \left( \mathbf{U}^f \left( \mathbf{X}^{(i)}(t), t \right), \mathbf{V}^{(i)}, \rho_f, \nu_f, \rho_p, D_p \right), \quad (6.6)$$

where  $\mathbf{U}^f \left( \mathbf{X}^{(i)}(t), t \right)$  is the fluid velocity at the particle location,  $\rho_f$  and  $\nu_f$  are the fluid thermodynamic density and kinematic viscosity, respectively, and  $\rho_p$  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

$$\rho_f \alpha_f \left( \frac{\partial \langle \mathbf{U}^f \rangle}{\partial t} + \langle \mathbf{U}^f \rangle \cdot \nabla \langle \mathbf{U}^f \rangle \right) = \nabla \cdot \langle \boldsymbol{\tau} \rangle - \langle \mathbf{F}^{\text{fp}} \rangle + \nabla \cdot \boldsymbol{\tau}^{RS}, \quad (6.7)$$

where  $\alpha_f$  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,  $\langle \mathbf{F}^{\text{fp}} \rangle$ , that appears in the fluid–phase mean momentum conservation equation is obtained from  $\langle \mathbf{f} \mid \mathbf{x}, \mathbf{v}; t \rangle$ , the conditional expectation of the force acting on the physical particles, as follows:

$$\langle \mathbf{F}^{\text{fp}} \rangle(\mathbf{x}, t) = \int_{[\mathbf{v}]} \langle \mathbf{f} \mid \mathbf{x}, \mathbf{v}; t \rangle f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \quad (6.8)$$

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,  $f_{\mathbf{V}}^c(\mathbf{v} \mid \mathbf{x}; t)$  (Subramaniam, 2001):

$$f(\mathbf{x}, \mathbf{v}, t) = n_p(\mathbf{x}, t) f_{\mathbf{V}}^c(\mathbf{v} \mid \mathbf{x}; t). \quad (6.9)$$

Spatial nonuniformity 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)

$$\frac{\partial n_p(\mathbf{x}, t)}{\partial t} + \nabla \cdot \{ \langle \mathbf{V}(\mathbf{x}, t) \rangle n_p(\mathbf{x}, t) \} = 0, \quad (6.10)$$

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

$$\frac{\partial \langle N_p(t) \rangle}{\partial t} = 0. \quad (6.11)$$

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. 6.4 and 6.10, the evolution equation for  $f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)$  (Subramaniam, 2001) is

$$\begin{aligned} \frac{\partial f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)}{\partial t} + \frac{\partial}{\partial x_k} [v_k f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t)] + \frac{\partial}{\partial v_k} \{ \langle A_k | \mathbf{x}, \mathbf{v}; t \rangle f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t) \} = \\ f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t) \frac{\partial \ln n_p(\mathbf{x},t)}{\partial x_k} \{ \langle \mathbf{V}(\mathbf{x},t) \rangle - v_k \} + f_{\mathbf{V}}^c(\mathbf{v}|\mathbf{x};t) \frac{\partial \langle V_k(\mathbf{x},t) \rangle}{\partial x_k}. \end{aligned} \quad (6.12)$$

As noted earlier, in LE simulations the physical system described by Eqs. 6.1 and 6.2 can be simulated with  $N_p$  real particles, or with  $N_c$  computational particles. Both simulations constitute an indirect solution of Eq. 6.4, or equivalently, of Eqs. 6.10 and 6.12. 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. 6.10, 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. 6.12.

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.

### 6.3 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.

### 6.3.1 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  $\{\mathbf{X}_c^{(i)}(t), \mathbf{V}_c^{(i)}(t), W^{(i)}, i = 1, \dots, N_c(t)\}$ , here  $\mathbf{X}_c^{(i)}(t)$  denotes the  $i^{\text{th}}$  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

$$\sum_{i=1}^{N_c(t)} W^{(i)} = \langle N_p(t) \rangle. \quad (6.13)$$

The position and velocity of the computational particles evolve by

$$\frac{d\mathbf{X}_c^{(i)}}{dt} = \mathbf{V}_c^{(i)} \quad (6.14)$$

$$\frac{d\mathbf{V}_c^{(i)}}{dt} = \mathbf{A}_c^{(i)}, \quad i = 1, \dots, N_c \quad (6.15)$$

where  $\mathbf{A}_c^{(i)}$  is the instantaneous acceleration experienced by the  $i^{\text{th}}$  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  $i$ -th 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

$$h(\mathbf{x}, \mathbf{v}, t) \equiv \langle h'(\mathbf{x}, \mathbf{v}, t) \rangle = \left\langle \sum_{i=1}^{N_c(t)} h_i'(\mathbf{x}, \mathbf{v}, t) \right\rangle = \left\langle \sum_{i=1}^{N_c(t)} W^{(i)} \delta_{\mathbf{X}_c^{(i)}} \delta_{\mathbf{V}_c^{(i)}} \right\rangle. \quad (6.16)$$

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.

### 6.3.2 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  $\{\mathbf{X}_c^{(i)}(t), \mathbf{V}_c^{(i)}(t), W^{(i)}(t), i = 1, \dots, N_c(t)\}$ . 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. 6.14 and 6.15, respectively. The statistical weights evolve by

$$\frac{dW^{(i)}(t)}{dt} = -\Omega^{(i)}(t)W^{(i)}(t) \quad , \quad i = 1, \dots, N_c(t), \quad (6.17)$$

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. 6.16, except that here the statistical weights  $W^{(i)}(t)$  are not constant but evolve in time. Similar to the decomposition of  $f$  in Eq. 6.9,  $h$  is decomposed as

$$h(\mathbf{x}, \mathbf{v}, t) = \tilde{n}_p(\mathbf{x}, t) \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x}; t), \quad (6.18)$$

where  $\tilde{h}_{\mathbf{V}_c}^c$  is the conditional velocity pdf of computational particles (the counterpart of  $f_{\mathbf{V}}^c$ ) and  $\tilde{n}_p(\mathbf{x}, t)$  is the physical number density implied by the present statistical description. The implied physical number density  $\tilde{n}_p(\mathbf{x}, t)$ , 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  $\langle W|\mathbf{x}; t \rangle$ :

$$\tilde{n}_p(\mathbf{x}, t) = \int_{[\mathbf{v}]} h d\mathbf{v} = \left\langle \sum_{i=1}^{N_c(t)} W^{(i)}(t) \delta(\mathbf{X}_c^{(i)}(t) - \mathbf{x}) \right\rangle = n_c(\mathbf{x}, t) \langle W|\mathbf{x}; t \rangle, \quad (6.19)$$

where the conditional expectation of the statistical weights is defined as

$$\langle W|\mathbf{x}; t \rangle = \frac{\left\langle W^{(i)}(t) \delta(\mathbf{X}_c^{(i)}(t) - \mathbf{x}) \right\rangle}{n_c(\mathbf{x}, t)}, \quad \text{if } n_c > 0, \quad (6.20)$$

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

We seek to guarantee the equivalence between  $h$  and  $f$  at all time by comparing evolution equations for  $h$ , and also the fundamental quantities ( $\tilde{n}_p, \langle N_p(t) \rangle, h_{\mathbf{V}_c}^c$ ) with their counterparts in the physical system. The evolution equation for  $h$  (for a detailed derivation, see Appendix D.1) is

$$\frac{\partial h}{\partial t} + \underbrace{\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle}_{\text{conditional expectation}} h + \frac{\partial}{\partial x_k} [v_k h] + \frac{\partial}{\partial v_k} \{ \langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle h \} = 0, \quad (6.21)$$

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

$$\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle = \frac{1}{h(\mathbf{x}, \mathbf{v}, t)} \left\langle \sum_{i=1}^{N_c} \left\{ \Omega^{(i)} h'_i(\mathbf{x}, \mathbf{v}, t) \right\} \right\rangle, \quad \text{if } h > 0 \quad (6.22)$$

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  $\tilde{n}_p$ , obtained by integrating Eq. 6.21 over  $\mathbf{v}$  space is (for a detailed derivaton, see Appendix D.3)

$$\frac{\partial \tilde{n}_p(\mathbf{x}, t)}{\partial t} + \nabla \cdot \{ \langle \mathbf{V}_c(\mathbf{x}, t) \rangle \tilde{n}_p(\mathbf{x}, t) \} = \underbrace{\langle \Omega | \mathbf{x}; t \rangle}_{\text{conditional expectation}} \tilde{n}_p(\mathbf{x}, t), \quad (6.23)$$

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

$$\langle \Omega | \mathbf{x}; t \rangle = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle}{\tilde{n}_p(\mathbf{x}, t)}, \quad \text{if } \tilde{n}_p > 0 \quad (6.24)$$

and equal to zero otherwise. The evolution equation for the total number of particles  $\langle N_p(t) \rangle$ , obtained by integrating Eq. 6.23 over  $\mathbf{x}$  space is (for a detailed derivation, see Appendix D.2)

$$\frac{\partial \langle N_p(t) \rangle}{\partial t} = - \underbrace{\langle N_p(t) \rangle}_{\text{total number}} \langle \Omega(t) \rangle, \quad (6.25)$$

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

$$\langle \Omega(t) \rangle = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \right\rangle}{\langle N_p(t) \rangle}. \quad (6.26)$$

From the evolution equations for  $h$  (Eq. 6.21) and number density  $\tilde{n}_p$  (Eq. 6.23), the evolution equation for  $\tilde{h}_{\mathbf{V}_c}^c$  can be obtained as (for a detailed derivation, see Appendix D.4)

$$\begin{aligned} \frac{\partial \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t)}{\partial t} + \frac{\partial}{\partial x_k} \left[ v_k \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t) \right] + \frac{\partial}{\partial v_k} \left\{ \langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t) \right\} = \\ - \underbrace{\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t) + \langle \Omega | \mathbf{x}; t \rangle \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t)}_{\tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t) \frac{\partial \ln \tilde{n}_p(\mathbf{x}, t)}{\partial x_k} \langle \mathbf{V}_c(\mathbf{x}, t) \rangle - v_k} + \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v}|\mathbf{x};t) \frac{\partial \langle V_{c,k}(\mathbf{x}, t) \rangle}{\partial x_k} \end{aligned} \quad (6.27)$$

### 6.3.2.1 Consistency requirements

In the above evolution equations (Eqs 6.21, 6.23, 6.25, and 6.27) for  $h$ ,  $\tilde{n}_p$ ,  $\langle N_p(t) \rangle$ , and  $\tilde{h}_{\mathbf{V}_c}^c$ , the underbraced quantities are the extra terms that appear when compared with the corresponding evolution equations (Eqs. 6.4, 6.10, 6.11, and 6.12) 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. 6.21 as compared to Eq. 6.4 for  $f$ ) is zero. This automatically then guarantees equivalence of the corresponding number density ( $\tilde{n}_p \equiv n_p$ ) and velocity PDF's ( $\tilde{h}_{\mathbf{V}_c}^c \equiv f_{\mathbf{V}_c}^c$ ). In summary, the computational particles and the real particles are statistically equivalent if the conditional (Eqs. 6.22 and 6.24), and unconditional (Eq. 6.26) expectations of fractional rate of change of statistical weight  $\Omega$  are all zero. These conditions on  $\Omega$  are consistency requirements, i.e., a prescription of  $\Omega^{(i)}(t)$  in Eq. 6.17 that satisfies these conditions guarantees that the evolution of computational particles by Eqs. 6.14, 6.15, and 6.17 corresponds to the evolution of the physical system as given by Eq. 6.4. 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.

### 6.3.2.2 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 (Kannenbergh 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. 6.19 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  $N_{pc}^T$ ) 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  $N_{pc}^T$  is a better alternative. In our simulations the minimum number of particles in each cell is specified to be  $0.5 N_{pc}^T$ , while the maximum number of particles allowed in each cell is  $2.0 N_{pc}^T$ . 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  $[0.5 N_{pc}^T, 2.0 N_{pc}^T]$ , the following actions are taken:

1.  $N_{pc} > 2.0 N_{pc}^T$ : 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 reduce to the desired value of  $2 N_{pc}^T$ .
2.  $N_{pc} < 0.5 N_{pc}^T$ : 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 <sup>3</sup> such as, velocity,

<sup>3</sup>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.

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

In the following we show that this number density control algorithm satisfies the consistency requirements as described in Sec. 6.3.2.1. 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. 6.14 and 6.15, 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. 6.17, but we omit the formal mathematical definition of  $\Omega^{(i)}(t)$  in favor of the easily understood algorithm.

#### 6.4 Numerical Estimation of Mean Interphase Momentum Transfer Term

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

$$\{\mathbf{F}_m^{fp}\} = \frac{1}{\mathcal{V}_m} \sum_{i=1}^{N_c(t)} \mathbf{f}_c^{(i)} W^{(i)} K(\mathbf{X}_c^{(i)}, \mathbf{x}^m), \quad (6.28)$$

where  $\mathbf{f}_c^{(i)} = m_p \mathbf{A}_c^{(i)}$  is the force acting on the  $i^{\text{th}}$  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  $\mathcal{V}_m$  is the geometric volume of the  $m^{\text{th}}$  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 by Fig. 6.2, the numerical estimate for  $\{F_{x,m}^{\text{fp}}\}$ , from Eq. 6.28, at the  $m^{\text{th}}$  grid node for an  $\mathcal{O}^{\text{th}}$ -order interpolation scheme is

$$\{F_{x,m}^{\text{fp}}\}^{\text{CE}} = \frac{1}{\mathcal{V}_m} \sum_{v=m-\mathcal{O}/2}^{m+\mathcal{O}/2-1} \sum_{k=1}^{N_c^v} f_x^k W^k b_l^x(\xi_l^k), \quad (6.29)$$

where  $l = m - v + \mathcal{O}/2$ ,  $b_l^x$  is the basis function at the elemental coordinate  $\xi_l^k$ ,  $N_c^v$  is the number of computational particles in the  $v^{\text{th}}$  cell,  $f_x^k$  is the  $x$  component of the force  $\mathbf{f}_c^{(k)}$  acting on the  $k^{\text{th}}$  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 5<sup>th</sup> 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

$$\{F_{x,m}^{\text{fp}}\}^{\text{IE}} = \frac{1}{\mathcal{V}_m} \frac{\sum_{v=m-\mathcal{O}/2}^{m+\mathcal{O}/2-1} \sum_{k=1}^{N_c^v} \phi_x^k b_l^x(\xi_l^k)}{\sum_{v=m-\mathcal{O}/2}^{m+\mathcal{O}/2-1} \sum_{k=1}^{N_c^v} b_l^x(\xi_l^k)}, \quad (6.30)$$

where  $\phi_x^k$  is a scaled force acting on the  $k^{\text{th}}$  particle in cell  $v$ , and superscript ‘IE’ stands for improved estimator. For the  $k^{\text{th}}$  particle belonging to the  $v^{\text{th}}$  cell,  $\phi_x^k$  is

$$\phi_x^k = f_x^k \sum_{j=1}^{N_c^v} W^j = f_x^k N_p^v, \quad (6.31)$$

where  $N_p^v$  is the number of physical particles in the  $v^{\text{th}}$  cell. On substituting the above expression for  $\phi_x^k$  into the expression for improved estimator (Eq. 6.30), we get

$$\{F_{x,m}^{\text{fp}}\}^{\text{IE}} = \frac{1}{\mathcal{V}_m} \sum_{v=m-\mathcal{O}/2}^{m+\mathcal{O}/2-1} \sum_{k=1}^{N_c^v} f_x^k W^{*k} b_l^x(\xi_l^k), \quad (6.32)$$

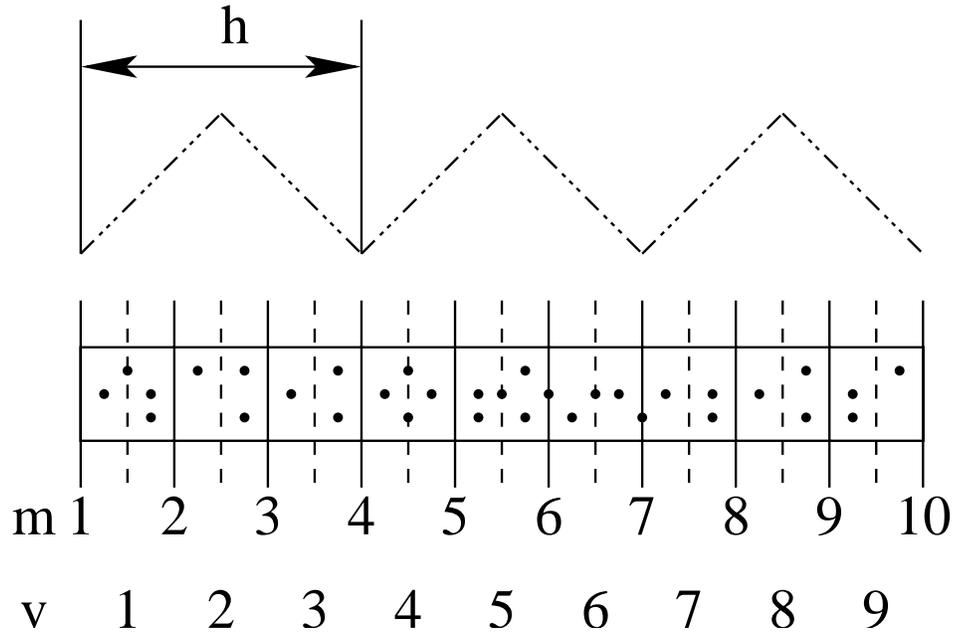


Figure 6.2 Schematic of a 1-D grid with dispersed phase particles shown by black dots. Solid and dashed vertical lines, indexed by  $m$ , show coarse and fine grids, respectively. The angled intersecting lines on the top represent a typical top hat kernel having bandwidth equal to  $h$ .

where  $W^{*k}$ , the effective statistical weight associated with the  $k^{\text{th}}$  particle is

$$W^{*k} = \frac{\sum_{j=1}^{N_c^v} W^j}{\sum_{v=m-O/2}^{m+O/2-1} \sum_{k=1}^{N_c^v} b_l^x(\xi_l^k)} = \frac{N_p^v}{\{N_{c,m}\}} \quad (6.33)$$

In the above expression,  $\{N_{c,m}\}$  is the effective number of computational particles at the  $m^{\text{th}}$  grid node. Therefore,  $W^{*k}$  can be interpreted as the locally averaged statistical weight. The expressions for the conventional (Eqs. 6.29) and the improved estimators (6.32) 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. 6.33.

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.

For the purpose of comparing the two estimators, we compare estimates of the mean interphase momentum transfer for a very simple case. The fluid velocity field  $\mathbf{U}^f = \{U_1^f, 0, 0\}$  is chosen to be of a simple transcendental form

$$U_1^f(x, y) = \cos\left(\frac{2\pi x}{\mathcal{L}_x}\right) \cos\left(\frac{2\pi y}{\mathcal{L}_y}\right), \quad (6.34)$$

in a domain  $\mathcal{D} = [0, \mathcal{L}_x] \times [0, \mathcal{L}_y] \times [0, \mathcal{L}_z]$ . The particles are uniformly distributed in the domain  $\mathcal{D}$ . The particle velocity  $\mathbf{V} = \{V_1, 0, 0\}$  is specified by the distribution of  $V_1$ . Two cases are considered for particle velocity distribution. In the first case, which is identical to the case described in original MS, a Gaussian velocity distribution with unit mean and zero variance is chosen, i.e.,  $V_1 \stackrel{D}{=} \mathcal{N}[1.0, 0.0]$ . In the second case, again a Gaussian particle velocity distribution is chosen but with a finite variance, i.e.,  $V_1 \stackrel{D}{=} \mathcal{N}[1.0, 0.3]$ . Given these quantities, the mean interphase momentum transfer term is known analytically.

The ensemble-averaged summed mean square of the total error  $\hat{\epsilon}_F$  is defined as

$$\hat{\epsilon}_F = \frac{1}{\mathcal{M}} \sum_{i=1}^{\mathcal{M}} \frac{\sum_{m=1}^M \left( \{\mathbf{F}_m^{fp}\}^i - \langle \mathbf{F}_m^{fp} \rangle \right)^2}{M}, \quad (6.35)$$

where  $\mathcal{M}$  is the number of independent realizations, and  $M = M_x M_y M_z$  is the total number of grid cells. The convergence characteristics of  $\hat{\epsilon}_F$  with number of particles per cell  $N_{pc}$  for the two cases of zero and non-zero variance in particle velocity distribution are shown by Figs. 6.3(a) and 6.3(b), respectively. The estimates have been obtained by LPI-2 interpolation scheme on a  $41 \times 41 \times 4$  and averaged over 100 independent realizations  $\mathcal{M}$ . It can be seen from Fig. 6.3(a) that, for the case of zero variance in particle velocity distribution, the improved estimator is more accurate than the conventional estimator by over two orders of

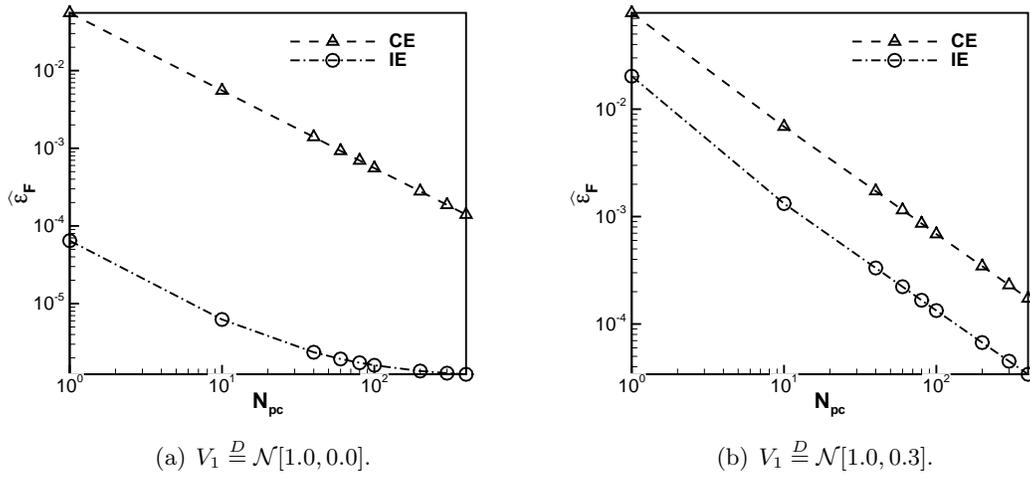


Figure 6.3 Convergence of Ensemble-averaged summed mean square error  $\hat{\epsilon}_F$  with number of particles per cell  $N_{pc}$  for (a) zero variance, and (b) finite variance in particle velocity distribution. The estimates have been obtained on a  $40 \times 40 \times 3$  grid using LPI-2 interpolation scheme. Number of independent realizations  $\mathcal{M}$  is equal to 100.

magnitude. Even with 1 particle per cell, improved estimator yields better estimates than those from conventional estimator with 100 particles per cell. However, for the case of finite variance in particle velocity distribution as shown by Fig. 6.3(b), the difference in accuracies (with improved estimator performing better than the conventional estimator) between the two estimators is reduced to approximately one order of magnitude.

The contour plot of root mean square statistical error  $\Sigma_F$ , computed for each estimator by a standard procedure (Xu and Pope, 1999; Garg et al., 2007), for the two cases of zero and non-zero variance in particle velocity distribution is shown by Figs 6.4(a) and 6.4(b), respectively. The number of particles per cell  $N_{pc}$  and number of independent realizations  $\mathcal{M}$  are equal to 5 and 100, respectively. The left and right panels in each subfigure correspond to the conventional and the improved estimator, respectively. From the two figures, it can be seen that for the case of non-zero variance in particle velocity distribution, the difference in the statistical errors between the estimators is the most. Statistical variations in an any estimate between the two realizations arises from variations in particle positions and as well

as particle velocities. Improved estimator (due to the nice scaling of terms in the numerator and denominator) has the ability to reduce statistical variations arising out of the particle positions. In the event of finite particle velocity variance, the improved estimator, compared to the conventional estimator, will yield lower total statistical error due to relatively lesser contribution from the first component (i.e., particle positions) of statistical error. For the case of zero particle velocity variance, only the uncertainties from particle positions contribute to the statistical error, and, therefore, we observe improved estimator to be more accurate than improved estimator by over two orders of magnitude. However, for the case of finite particle velocity variance, statistical error from the second component (i.e., from variation in the particle velocities) is the dominant term. Since the improved estimator results in lesser statistical error from particle position variation, and the second statistical error component is constant for both the estimators, improved estimator, although not on the same order as in the first case, still performs better than the conventional estimator. From the above comparison, the superior performance of the improved estimator is attributed to its ability in incurring lesser statistical error arising due to particle position variations.

## 6.5 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. 6.7) 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  $L_{\text{ref}}$  and lid velocity  $U_{\text{ref}}$ , 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

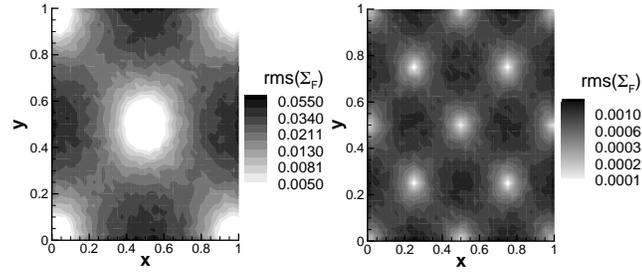
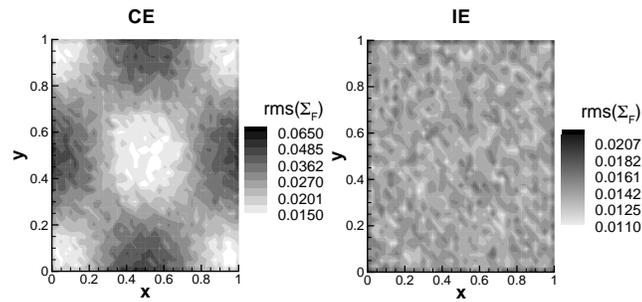
(a)  $V_1 \stackrel{D}{=} \mathcal{N}[1.0, 0.0]$ .(b)  $V_1 \stackrel{D}{=} \mathcal{N}[1.0, 0.3]$ .

Figure 6.4 Contour plot of the root mean square statistical error  $\Sigma_F$  for the two cases of (a) zero, and (b) non-zero variance in particle velocity distribution. The number of particles per cell  $N_{pc}$  and number of independent realizations  $\mathcal{M}$  are equal to 5 and 100, respectively. The left and right panels in each subfigure correspond to the conventional and the improved estimator, respectively.

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

$$\mathbf{A}^{*(i)} = \frac{\mathbf{U}^f(\mathbf{X}^{(i)}, t) - \mathbf{V}^{(i)}}{\tau_p}, \quad (6.36)$$

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  $\alpha_p$  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  $50 \times 50$  grid cells to the finest resolution of  $100 \times 100$  grid cells.

For the TLE simulation, two different approaches are used. In the first approach, referred to as TLE1, real particles ( $\approx 5300$ ) 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,  $N_{pc}(t = 0) = 20$ , are uniformly seeded at the beginning of the simulation.

For the ILE simulation, the computational particles are initially seeded as in TLE2, i.e.,  $N_{pc}(t = 0) = 20$ . 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

$$\epsilon_F = \frac{1}{M} \sqrt{\sum_{m=1}^M \left( \frac{\{\mathbf{F}_m^{fp}\} - \langle \mathbf{F}_m^{fp} \rangle}{\langle \mathbf{F}_m^{fp} \rangle} \right)^2}, \quad (6.37)$$

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  $150 \times 150$  grid is taken to be the reference solution for the purpose of error calculation. The relative root mean square error  $\epsilon_F$  for each grid is calculated by substituting the interpolated value of reference solution for  $\langle \mathbf{F}_m^{fp} \rangle$  in the above equation.

The particle-laden lid-driven cavity problem is simulated for 10 non-dimensional time units ( $t^* = t/\tau_f$ ). Figure 6.1 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. Figure 6.5 compares the convergence characteristics of the root mean square relative error  $\epsilon_F$ , 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.

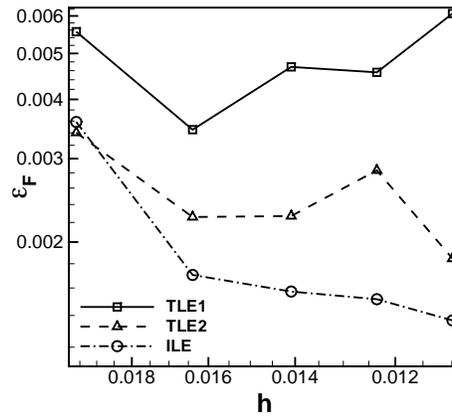


Figure 6.5 Comparison of the convergence characteristics of the root mean square relative error  $\epsilon_F$  with grid spacing  $h$  for TLE1, TLE2, and ILE simulations of the lid-driven cavity flow. Lines are a simple fit to the data points.

## 6.6 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

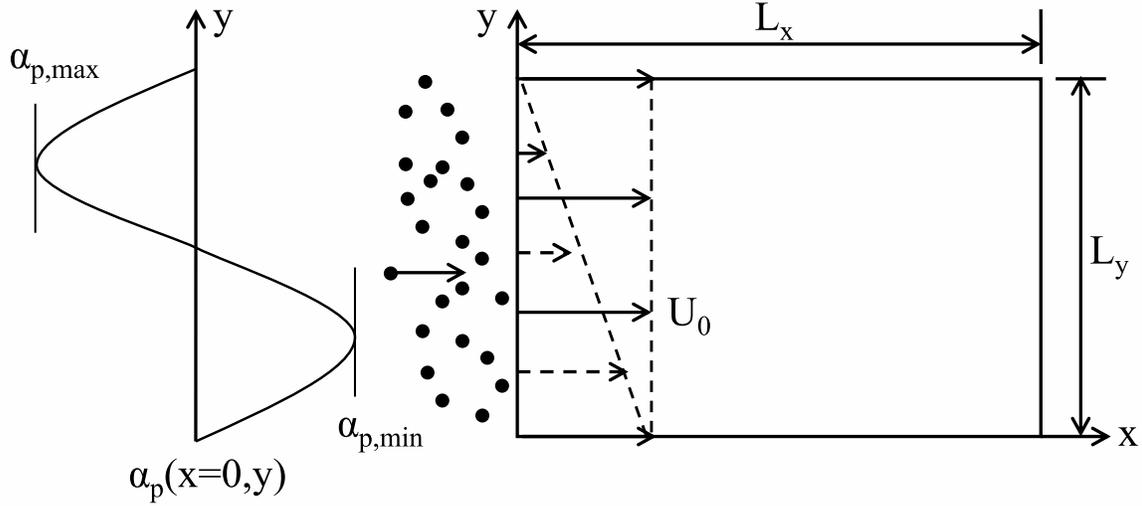


Figure 6.6 Schematic of the test problem. Solid line vectors represent  $U_1^f$ , dashed line vectors represent  $U_2^f$ . and particles, injected at  $x = 0$ , are shown as black dots.

particle and the fluid is  $\mathcal{O}(1)$ . Therefore, the linear drag model given by Eq. 6.36 is valid here also.

As represented by the schematic in Fig. 6.6, a frozen two-dimensional fluid velocity field

$$U_1^f(x, y) = U_0, \quad (6.38)$$

$$U_2^f(x, y) = U_0 \left( 1 - \frac{y}{L_y} \right), \quad (6.39)$$

is chosen in a domain  $\mathcal{D} = [0, L_x] \times [0, L_y]$ . For this fluid velocity field, the flow time scale  $\tau_f$  is defined to be equal to  $L_y/U_0$ , 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. 6.14 and 6.15) can be reduced to two second-order ordinary differential equations by substituting the frozen fluid velocity field (Eqs. 6.38 and 6.39) into the particle acceleration model (Eq. 6.36). 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 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. 6.1 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

$$X_2(t, X_{2,0}) = \exp -t/2\tau_p [X_{2,0} - \mathcal{L}_y] \left\{ \cos \left( \frac{\zeta t}{2\tau_p} \right) + \frac{1}{\zeta} \sin \left( \frac{\zeta t}{2\tau_p} \right) \right\} + \mathcal{L}_y, \quad (6.40)$$

and

$$V_2(t, X_{2,0}) = \frac{2U_0 \left[ 1 - \frac{X_2}{L_y} \right] \sin \left( \frac{\zeta t}{2\tau_p} \right)}{\zeta \left\{ \cos \left( \frac{\zeta t}{2\tau_p} \right) + \frac{1}{\zeta} \sin \left( \frac{\zeta t}{2\tau_p} \right) \right\}}, \quad (6.41)$$

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  $X_1(t, X_{1,0}) = X_{1,0} + U_0t$ .

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. 6.41) by the transformation

$$\mathbf{U}^p(\mathbf{X}(t, \mathbf{X}_0), t) = \mathbf{V}(t, \mathbf{X}_0). \quad (6.42)$$

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

$$\alpha_p(x = 0, y) = \frac{\alpha_{p,\max} + \alpha_{p,\min}}{2} + \frac{\alpha_{p,\max} - \alpha_{p,\min}}{2} \sin \left( \frac{2\pi y}{\mathcal{L}_y} \right), \quad (6.43)$$

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. 6.40), 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

$$\alpha_p(x, y) = \begin{cases} \alpha_p(x=0, Y^{-1}(y))e^{\hat{x}}/g(\hat{x}) & : \quad 0 < x \leq \mathcal{L}_x, \\ & \mathcal{L}_y \{1 - e^{\hat{x}}g(\hat{x})\} < y \leq \mathcal{L}_y; \\ 0 & : \quad \text{otherwise,} \end{cases} \quad (6.44)$$

where  $Y^{-1}(y)$  is an inverse function obtained by re-expressing  $X_{2,0}$  in Eq. 6.40 in terms of  $X_2$ , such that

$$Y^{-1}(y) \equiv X_{2,0} = \frac{y - \mathcal{L}_y}{g(\hat{x})} \exp^{\hat{x}} + \mathcal{L}_y, \quad (6.45)$$

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  $n_p(x, y) = \frac{\alpha_p(x, y)}{\mathcal{V}_p}$ , where  $\mathcal{V}_p$  is the particle volume. Fig. 6.7(a) 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. 6.44), it is noted that in the region bounded by  $0 < x \leq \mathcal{L}_x$  and  $0 \leq y \leq \mathcal{L}_y \{1 - e^{\hat{x}}g(\hat{x})\}$ , 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. 6.8 is

$$\langle \mathbf{F}^{\text{fp}} \rangle(\mathbf{x}) = \frac{m_p n_p(\mathbf{x})}{\tau_p} \left[ \mathbf{U}^f(\mathbf{x}) - \mathbf{U}^p(\mathbf{x}) \right]. \quad (6.46)$$

Figure 6.7(b) shows the contour plot of the scaled analytical mean interphase momentum transfer term in  $y$ -direction obtained after substituting the fluid velocity field (Eqns. 6.38 and 6.39), number density field, and mean particle velocity field (Eq. 6.42) into Eq. 6.46. 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. 6.36), (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

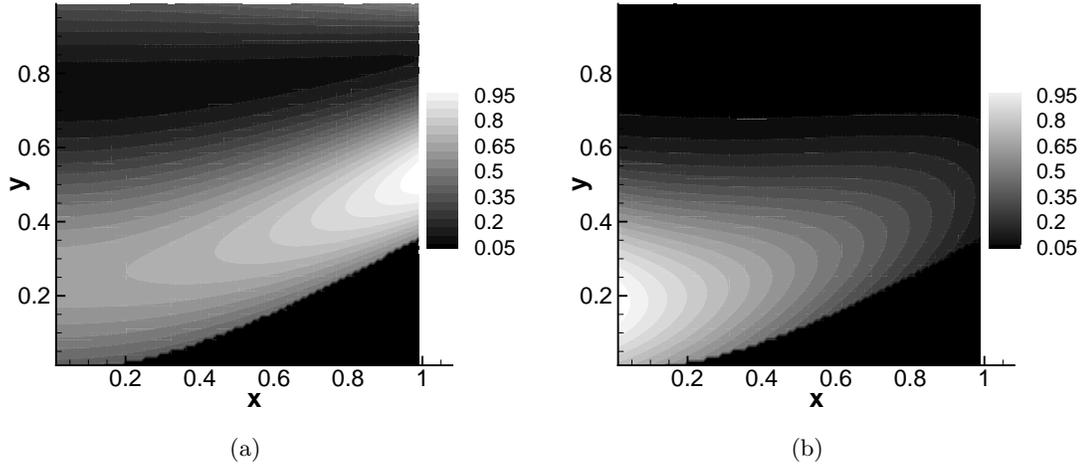


Figure 6.7 Contour plots of scaled analytical mean (a) number density  $\langle n_p \rangle / \langle n_p \rangle_{\max}$  and (b) interphase momentum transfer term  $\langle F_y^{\text{fp}} \rangle / \langle F_y^{\text{fp}} \rangle_{\max}$  for the test problem.

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.

### 6.6.1 Computation setup

The physical domain  $\mathcal{D}$  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 with  $29 \leq M_x = M_y \leq 99$ , 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. 6.43)  $\alpha_{p,\max} = 0.01$ , 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  $\mathcal{V}^{\text{in}} = U_0 \Delta t \Delta y \Delta z$ , such that the number of real particles introduced at each time step in  $c^{\text{th}}$  cell that adjoins the boundary at  $x = 0$  is given by

$$N_p^{\text{in}}(y_c) = \left\lceil \frac{\alpha_p(x=0, y_c) \mathcal{V}^{\text{in}}}{\mathcal{V}_p} \right\rceil, \quad (6.47)$$

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  $N_p^{\text{in}}$  computation in the next time step. These  $N_p^{\text{in}}(y_c)$  particles are uniformly distributed in the volume  $\mathcal{V}^{\text{in}}$ .

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  $N_c^{\text{in}}$  in the  $c^{\text{th}}$  cell is computed as

$$N_c^{\text{in}}(y_c) = \lceil n_c^{\text{in}} \mathcal{V}^{\text{in}} \rceil, \quad (6.48)$$

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

$$W^{(i)} = \frac{N_p^{\text{in}}}{N_c^{\text{in}}}, \quad i = 1, \dots, N_c^{\text{in}}. \quad (6.49)$$

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. 6.44). 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 parameters that control the computational particle distribution in TLE2 is  $N_{pc}^{in}$ , and for TLE, it is  $N_{pc}^T$ . 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 are equal to 690,000 and 747,282, respectively. These values are obtained with  $N_{pc}^{in} = 23$  and  $N_{pc}^T = 20$ . In this way, a comparable numerical resolution is maintained between TLE2 and ILE simulations.

## 6.7 Results

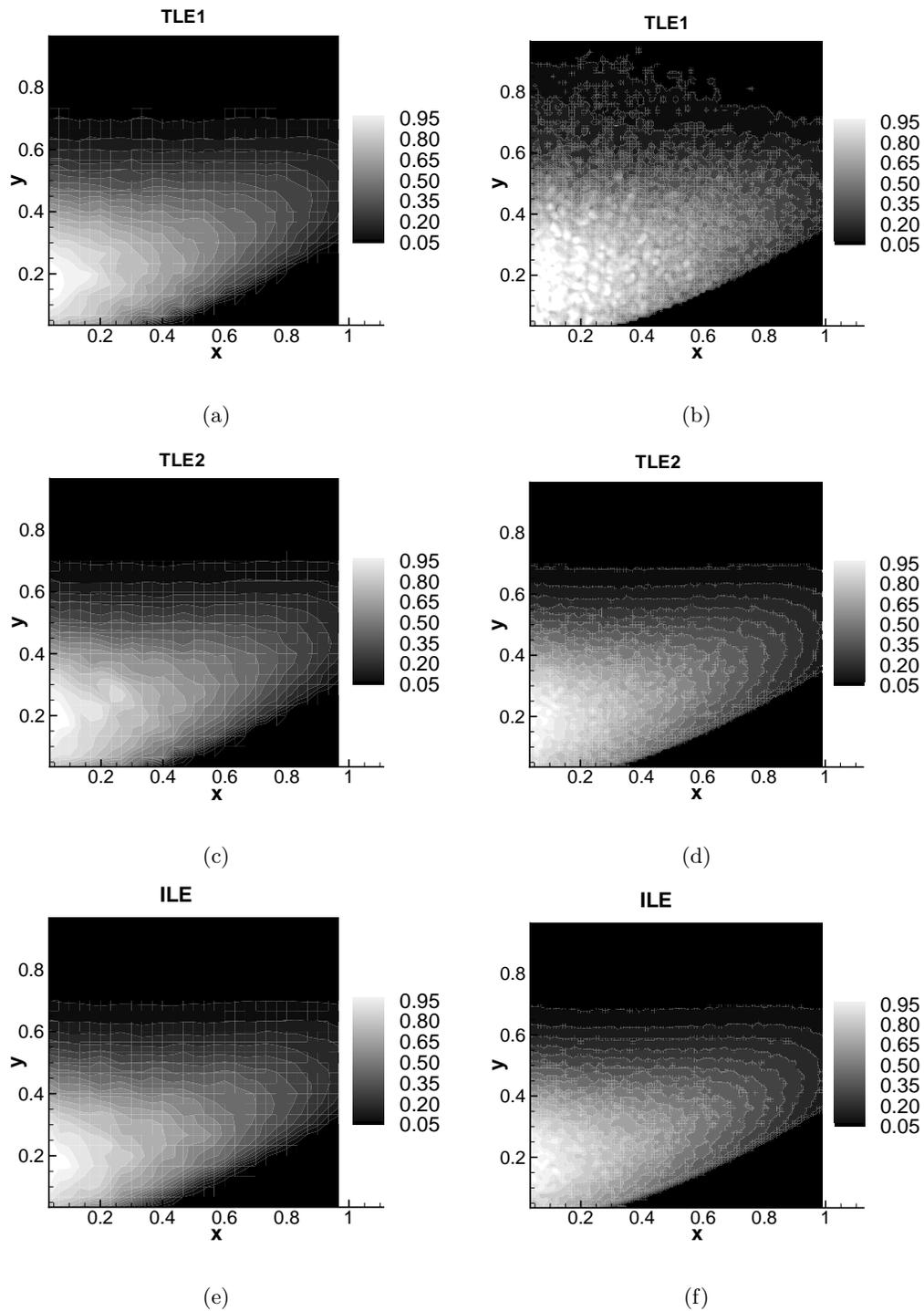


Figure 6.8 Contour plots of the scaled mean interphase momentum transfer term  $\{F_y^{\text{fp}}\} / \langle F_y^{\text{fp}} \rangle_{\text{max}}$  obtained from TLE1, TLE2, and ILE

simulations of the test problem on  $30 \times 30 \times 3$  ((a), (c), and (e)) and  $100 \times 100 \times 3$  ((b), (d), and (f)) grids using LPI-2 with improved estimator.

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  $\{\mathbf{F}^{\text{fp}}\}$  and its relative error obtained using LPI-2 are reported here. Figures 6.8(a), 6.8(c), and 6.8(e) show, respectively, the contour plots of  $\{F_y^{\text{fp}}\}/\langle F_y^{\text{fp}} \rangle_{\text{max}}$  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. 6.8(b)), TLE2 (Fig. 6.8(d)) and ILE (Fig. 6.8(f)) simulations, clearly show the worsening of estimates for the TLE1 although TLE2 and ILE do not give very different estimates.

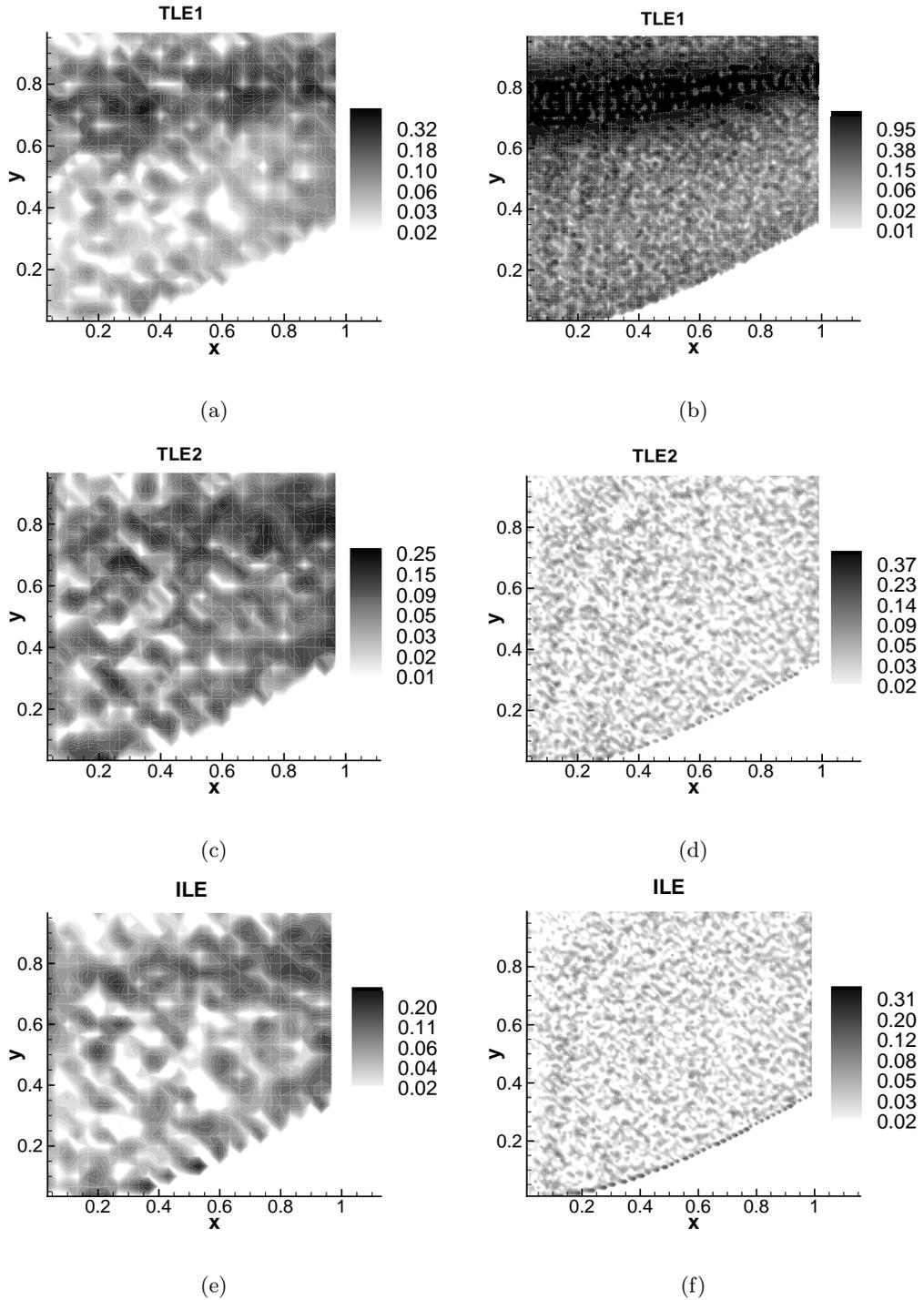


Figure 6.9 Contour plots of relative error  $\left( \left| \frac{\langle F_y^{\text{fp}} \rangle - \{F_y^{\text{fp}}\}}{\langle F_y^{\text{fp}} \rangle} \right| \right)$  obtained from TLE1, TLE2, and ILE simulations on  $30 \times 30 \times 3$  ((a), (c), and (e)) and  $100 \times 100 \times 3$  ((b), (d), and (f)) grids using LPI-2 with improved estimator.

Figures 6.9(a), 6.9(c), and 6.9(e) show, respectively, the contour plots of relative error  $\left( \left| \frac{\langle F_y^{\text{fp}} \rangle - \{F_y^{\text{fp}}\}}{\langle F_y^{\text{fp}} \rangle} \right| \right)$  from TLE1, TLE2, and ILE simulations on the coarsest grid. The relative error for the finest grid resolution is shown in Figs. 6.9(b) (TLE1), 6.9(d) (TLE2), and 6.9(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. 6.9(a)) the relative error is nearly uniform over the entire domain. However, on the  $100 \times 100 \times 3$  grid (Fig. 6.9(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  $0.0 < x < 1.0$ ).

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 nonuniform 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. Figure 6.10 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. 6.11 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

	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

Table 6.1 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.

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 linear least-squares fit to the data in Figs. 6.10 and 6.11. The convergence rates are summarized in Table 6.1 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 6.1, 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.

## 6.8 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; Jaber 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 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 does not change drastically in time. While we do not anticipate any special difficulties with simulating such flows, they may be a 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. 6.30 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. 6.30 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. 6.2 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<sup>4</sup> in the estimate scales as a power of the bandwidth, independent of the grid-size.

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

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.

## 6.9 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 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.

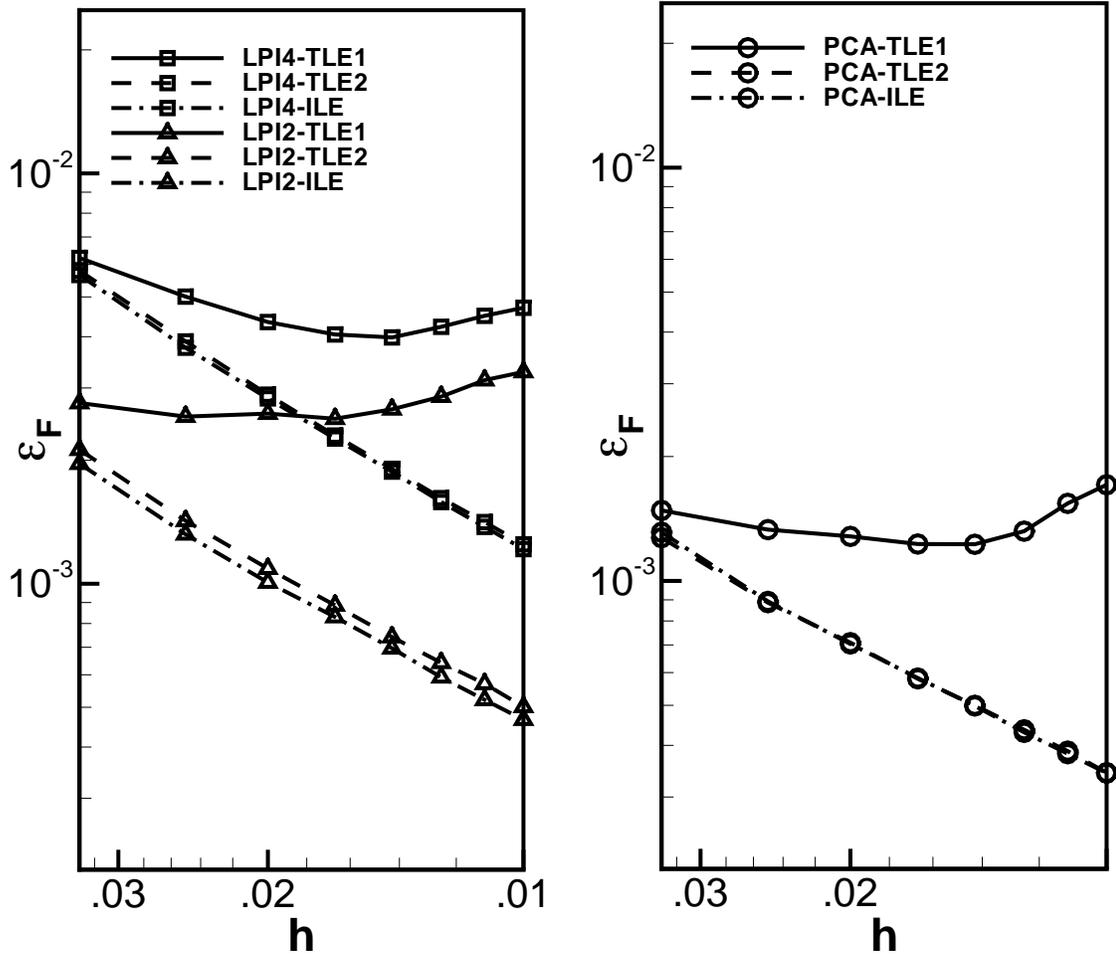


Figure 6.10 Convergence characteristics of the root mean square relative error with grid spacing  $h$  for TLE1 (solid), TLE2 (dash), and ILE (dash-dot) simulations of the test problem. Conventional estimator is used. Lines are simple fit to the symbols.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA.

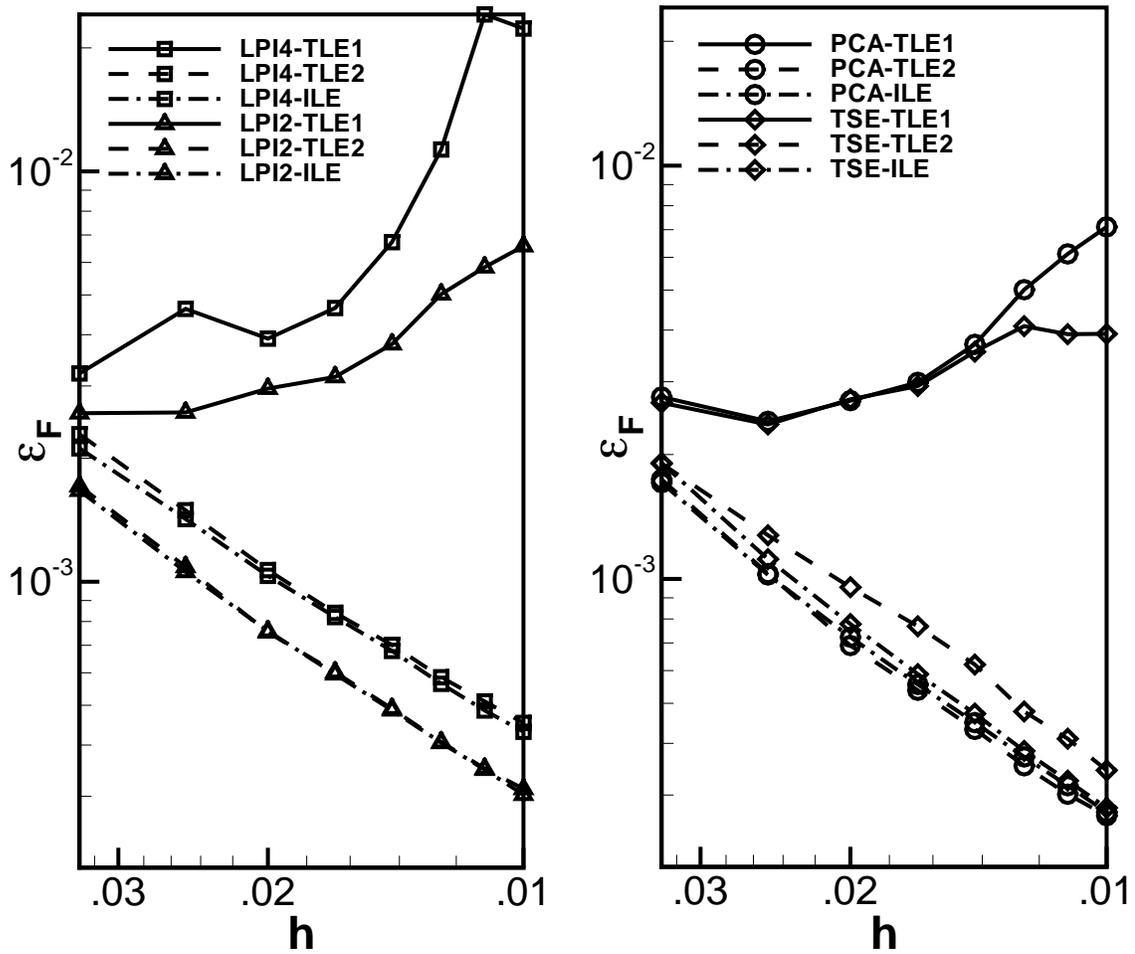


Figure 6.11 Convergence characteristics of the root mean square relative error with grid spacing  $h$  for TLE1 (solid), TLE2 (dash), and ILE (dash-dot) simulations of the test problem. Improved estimator is used. Lines are simple fit to the symbols.  $\square$ , LPI-4;  $\triangle$ , LPI-2;  $\circ$ , PCA,  $\diamond$ , TSE.

## CHAPTER 7. DEVELOPMENT OF AN OPEN SOURCE DEM CODE FOR SIMULATIONS OF GRANULAR AND GAS-PARTICLE SYSTEMS

This chapter is a manuscript in preparation titled “Development of an open source DEM code for simulations of granular and gas-particle systems” lead by R. Garg, in collaboration with co-workers at National Energy Technology Laboratory (NETL), Morgantown, WV, and S. Subramaniam. An extended version of this manuscript will appear as a theory guide and user manual for the MFI<sub>X</sub>-CDM code on “[www.mfix.org](http://www.mfix.org)”.

In the MFI<sub>X</sub> code, a basic structure for DEM (Discrete Element Method) simulations has existed for few years, termed MFI<sub>X</sub>-CDM, where CDM stands for continuum-discrete model. We purposely term it this way as it is easy to confuse DEM for purely granular flows versus that where DEM is coupled with continuum description for gas. Here the gas-phase is solved using continuum equations while the solids phase is solved using discrete particles. However, MFI<sub>X</sub> is not as widely used for CDM simulations as it is for continuum simulations even though it is an excellent opportunity to be able to run different descriptions from one platform. We suspect that the main reason behind MFI<sub>X</sub>-CDM modules lesser usage is its lack of reliability partly due to limited verification and validation. This is in spite of the fact that two theses (Boylakunta, 2003; Weber, 2004), several publications (Weber and Hrenya, 2006; Sun et al., 2007) and the previous MFI<sub>X</sub>-DEM document (Boylakuntla and Pannala, 2006). The main reason has been that the gas-solids flows are extremely complex and for the systems studied, the inaccuracies in the formulation did not seem to have had an effect on the qualitative and semi-quantitative validation exercise.

The MFI<sub>X</sub>-CDM code has recently been extensively debugged and two new features have been added: 1) the ability to simulate dilute systems, and 2) the ability to accommodate any

particle size/density distribution. In order to allow it to accurately simulate dilute systems, a suite of interpolation routines have been added to calculate the drag force on each particle and also its reverse projection onto the Eulerian grid.

### 7.1 Theory: Introduction and Background

The simulation technique wherein the solids is represented by discrete particles is referred to as the discrete element method (DEM). DEM simulations are used frequently for simulating granular flows and also gas–particle systems. In the alternative continuum approach, constitutive relations are needed to model the stresses in solids phase. Since the collisions are resolved in DEM approach, no such models are needed in these simulations, although a force model for the particle interactions must be introduced. Therefore, DEM simulations, although computationally more expensive than continuum based simulations, serve as a good tool to verify any continuum constitutive model for solids phase stresses. They also provide insight that can aid in developing new models.

For gas–particle flows, there are many codes available, both commercial (Fluent) and open source (CFDlib, MFIX), that can solve the averaged continuum equations (i.e. both phases represented as continua). Since the discrete particle model currently available in Fluent considers only the stochastic collisions (O’Rourke and Amsden, 1987), it is therefore limited to describing dilute systems. Similarly, for granular flows, commercial and open-source codes are available. However, MFIX provides a single source code having the capability to solve equations both for the continuum and the discrete descriptions that can be used to simulate gas–particle systems, as well as, granular flows.

In the next section, the details of DEM simulations are provided in a manner that is consistent with the MFIX–CDM implementation.

### 7.2 Theory and Numerical Implementation

In the MFIX–CDM approach, the gas–phase governing equations for both continuity and mass conservation are similar to traditional gas–phase CFD with additional coupling terms due

to drag from the solids–phase. The solids–phase is modeled using discrete particles. Below we provide the list of governing equations along with the numerical implementation including the coupling procedure.

### 7.2.1 Gas-phase Computations

The governing equations, implemented in MFIx (Syamlal et al., 1993), for the gas–phase continuity and momentum conservation in the absence of growth, aggregation, and breakage phenomena are:

$$\frac{\partial(\varepsilon_g \rho_g)}{\partial t} + \nabla \cdot (\varepsilon_g \rho_g \mathbf{v}_g) = 0 \quad ; \quad (7.1)$$

and

$$\frac{D}{Dt}(\varepsilon_g \rho_g \mathbf{v}_g) = \nabla \cdot \bar{\bar{S}}_g + \varepsilon_g \rho_g \mathbf{g} - \sum_{m=1}^M \mathbf{I}_{gm} \quad . \quad (7.2)$$

In the above equation,  $\varepsilon_g$  is the gas–phase volume fraction,  $\rho_g$  is the thermodynamic density of the gas phase,  $\mathbf{v}_g$  is the volume–averaged gas–phase velocity,  $\mathbf{I}_{gm}$  is the momentum transfer term between the gas and the  $m^{\text{th}}$  solid phase, and  $\bar{\bar{S}}_g$  is the gas–phase stress tensor given by

$$\bar{\bar{S}}_g = -P_g \bar{\bar{I}} + \bar{\bar{\tau}}_g, \quad (7.3)$$

where  $P_g$  is the gas–phase pressure. Also,  $\bar{\bar{\tau}}_g$  is the gas–phase shear stress tensor,

$$\bar{\bar{\tau}}_g = 2\varepsilon_g \mu_g \bar{\bar{D}}_g + \varepsilon_g \lambda_g \nabla \cdot \text{tr}(\bar{\bar{D}}_g) \bar{\bar{I}}, \quad (7.4)$$

where  $\bar{\bar{D}}_g = \frac{1}{2} [\nabla \mathbf{v}_g + (\nabla \mathbf{v}_g)^T]$  is the strain rate tensor, and  $\mu_g$  and  $\lambda_g$  are the dynamic and second coefficients of viscosity of the gas phase. Solid phases are differentiated based according to radii and densities. Therefore, the diameter and density of the  $m^{\text{th}}$  solid–phase is denoted by  $D_m$  and  $\rho_{sm}$ , respectively.

### 7.2.2 Discrete Element Method: DEM

In the DEM approach,  $m^{\text{th}}$  solid–phase is represented by  $N_m$  spherical particles with each particle having diameter  $D_m$  and density  $\rho_{sm}$ . For a total of  $M$  solid phases, the total number of particles is equal to  $N = \sum_{m=1}^M N_m$ . These  $N$  particles are represented in a Lagrangian frame of

reference at time  $t$  by  $\{\mathbf{X}^{(i)}(t), \mathbf{V}^{(i)}(t), \boldsymbol{\omega}^{(i)}(t), D^{(i)}, \rho^{(i)} \mid i = 1, \dots, N\}$ , where  $\mathbf{X}^{(i)}(t)$  denotes the  $i^{\text{th}}$  particle's position,  $\mathbf{V}^{(i)}(t)$  and  $\boldsymbol{\omega}^{(i)}$  denote its linear and angular velocities,  $D^{(i)}$  denotes its diameter, and  $\rho^{(i)}$  represents its density. It is implicit that if a particle belongs to  $m^{\text{th}}$  solid-phase, then its diameter and density are, respectively, equal to  $D_m$  and  $\rho_{sm}$  (i.e., equal to the diameter and density of the  $m^{\text{th}}$  solid-phase). The mass  $m^{(i)}$  and moment of inertia  $I^{(i)}$  of the  $i^{\text{th}}$  particle are equal to  $\rho^{(i)} \frac{\pi D^{(i)3}}{6}$  and  $\frac{m^{(i)} D^{(i)2}}{10}$ , respectively. The position, linear and angular velocities of the  $i^{\text{th}}$  particle evolve according to Newton's laws as:

$$\frac{d\mathbf{X}^{(i)}(t)}{dt} = \mathbf{V}^{(i)}(t), \quad (7.5)$$

$$m^{(i)} \frac{d\mathbf{V}^{(i)}(t)}{dt} = m^{(i)} \mathbf{g} + \mathbf{F}_d^{(i \in k, m)}(t) + \mathbf{F}_c^{(i)}(t), \quad (7.6)$$

$$I^{(i)} \frac{d\boldsymbol{\omega}^{(i)}(t)}{dt} = \frac{1}{2} D^{(i)} \boldsymbol{\eta} \times \mathbf{F}_c^{(i)}(t), \quad (7.7)$$

$$(7.8)$$

where  $\mathbf{g}$  is the acceleration due to gravity,  $\mathbf{F}_d^{(i \in k, m)}$  is the total drag force (pressure + viscous) on the  $i^{\text{th}}$  particle residing in the  $k^{\text{th}}$  cell and belonging to the  $m^{\text{th}}$  solid-phase,  $\mathbf{F}_c^{(i)}$  is the net contact force acting as a result of contact with other particles, and  $\boldsymbol{\eta}$  is the outward pointing normal unit vector along the particle radius. The next two subsections discuss in detail the calculation of the contact and drag forces.

### 7.2.2.1 Contact Forces

The advantage of the DEM approach over that of solving the continuum equations for the solid-phase lies in explicit treatment of particle-particle collisions. For two-phase flows, hard-sphere (based on the event driven algorithm, first proposed by Allen and Tildesley (1989)) and soft-sphere (based on the spring-dashpot model, first proposed by Cundall and Strack (1978)) models are the two most commonly used approaches. In the hard-sphere approach, collisions are binary and instantaneous, whereas the soft-sphere approach imposes no such restriction since it is possible to have enduring, multi-particle contacts. In the event driven (hard-sphere) approach, the time step is determined by the minimum collision time between any one pair of particles — which is directly proportional to the mean free path or inversely

proportional to the particle volume fraction. Therefore, the hard-sphere approach is most suitable for dilute systems, since in denser systems the minimum collision time becomes much smaller than other time scales. Also, in dense regions, momentum transfer occurs more through enduring contacts (the so called quasi-static regime) than through binary collisions. Even in gas-particle systems that are nominally dilute, the preferential concentration of particles to the high strain rate regions of gas flow can result in locally dense regions which require very small time steps to resolve. The time step in the soft-sphere approach, although small and a function of the spring stiffness, does not vary with the volume fraction. Although the hard-sphere approach may be a good alternative in some systems, the soft-sphere approach is more robust due to the independence of time step size on volume fraction.

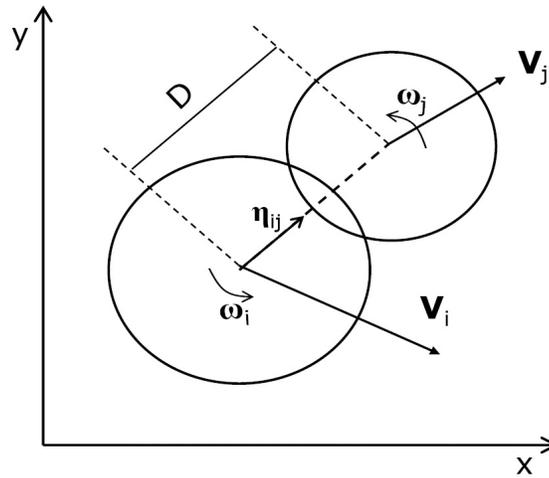


Figure 7.1 Schematic of two particles  $i$  and  $j$  having diameters  $D_i$  and  $D_j$  in contact. Particles have linear and angular velocities equal to  $\mathbf{V}_i, \mathbf{V}_j$  and  $\omega_i, \omega_j$ , respectively. Overlap  $\delta_{ij} = 0.5(D_i + D_j) - D$ .  $\eta_{ij}$  is the vector along the line of contact pointing from particle  $i$  to particle  $j$ .

Below the soft-sphere collision approach implemented in MFIx-CDM code is detailed. As shown by the schematic in Fig. 7.1, consider two particles  $i$  and  $j$  in contact that have diameters equal to  $D^{(i)}$  and  $D^{(j)}$  and are located at  $\mathbf{X}^{(i)}$  and  $\mathbf{X}^{(j)}$ . The particle  $i$  is moving with linear and angular velocities equal to  $\mathbf{V}^{(i)}$  and  $\omega^{(i)}$ , respectively. Similarly, the particle  $j$

is moving with linear and angular velocities equal to  $\mathbf{V}^{(j)}$  and  $\boldsymbol{\omega}^{(j)}$ , respectively. The normal overlap between the particles is calculated as

$$\delta_{ij} = 0.5 \left( D^{(i)} + D^{(j)} \right) - \left| \mathbf{X}^{(i)} - \mathbf{X}^{(j)} \right|. \quad (7.9)$$

The unit vector along the line of contact pointing from particle  $i$  to particle  $j$  is

$$\boldsymbol{\eta}_{ij} = \frac{\mathbf{X}^{(j)} - \mathbf{X}^{(i)}}{\left| \mathbf{X}^{(j)} - \mathbf{X}^{(i)} \right|}, \quad (7.10)$$

and the relative velocity of the point of contact becomes

$$\mathbf{V}_{ij} = \mathbf{V}^{(i)} - \mathbf{V}^{(j)} + \frac{1}{2} \left( D^{(i)} \boldsymbol{\omega}^{(i)} + D^{(j)} \boldsymbol{\omega}^{(j)} \right) \times \boldsymbol{\eta}_{ij}. \quad (7.11)$$

Therefore, the normal  $\mathbf{V}_{nij}$  and tangential  $\mathbf{V}_{tij}$  components of contact velocity, respectively, are

$$\mathbf{V}_{nij} = \mathbf{V}_{ij} \cdot \boldsymbol{\eta}_{ij} \boldsymbol{\eta}_{ij} \equiv \left( \mathbf{V}^{(i)} - \mathbf{V}^{(j)} \right) \cdot \boldsymbol{\eta}_{ij} \boldsymbol{\eta}_{ij}, \quad (7.12)$$

and

$$\mathbf{V}_{tij} = \mathbf{V}_{ij} - \mathbf{V}_{ij} \cdot \boldsymbol{\eta}_{ij} \boldsymbol{\eta}_{ij}. \quad (7.13)$$

The tangent to the plane of contact  $\mathbf{t}_{ij}$  is

$$\mathbf{t}_{ij} = \frac{\mathbf{V}_{tij}}{\left| \mathbf{V}_{tij} \right|}. \quad (7.14)$$

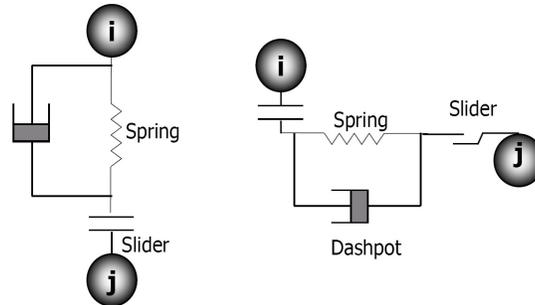


Figure 7.2 Schematic of the spring–dashpot system used to model particle contact forces in soft–sphere approach.

In the soft-sphere approach, the overlap between the two particles is represented as a system of springs and dashpots (Fig. 7.2) in both normal and tangential directions. The spring causes the rebound off the colliding particles and the dashpot mimics the dissipation of kinetic energy due to inelastic collisions. The spring stiffness coefficients in the tangential and normal directions are  $k_t$  and  $k_n$ , respectively. Similarly, the dashpot damping coefficients in the tangential and normal directions are  $\eta_t$  and  $\eta_n$ , respectively. The spring stiffness and dashpot damping coefficients are essentially a function of the solid-phases the colliding particles belong to. For example, if the  $i^{\text{th}}$  particle belongs to the  $m^{\text{th}}$  solid-phase and the  $j^{\text{th}}$  particle belongs to the  $\ell^{\text{th}}$  solid-phase, then the spring stiffness coefficients are given by  $k_{nm\ell}$  and  $k_{tm\ell}$ . Similarly, the dashpot damping coefficients are given by  $\eta_{nm\ell}$  and  $\eta_{tm\ell}$ . However, in order to keep the formulation simple, the subscripts  $(m, \ell)$  are dropped and it is noted that the spring stiffness and dashpot damping coefficients will depend on the solid-phases the colliding particles belong to.

The normal and tangential components of the contact force  $\mathbf{F}_{ij}$ , at time  $t$ , are decomposed into the spring (conservative) force  $\mathbf{F}_{ij}^{\text{S}}$  and the dashpot (dissipative) force  $\mathbf{F}_{ij}^{\text{D}}$  as

$$\mathbf{F}_{nij}(t) = \mathbf{F}_{nij}^{\text{S}}(t) + \mathbf{F}_{nij}^{\text{D}}(t), \quad (7.15)$$

and

$$\mathbf{F}_{tij}(t) = \mathbf{F}_{tij}^{\text{S}}(t) + \mathbf{F}_{tij}^{\text{D}}(t). \quad (7.16)$$

The dashpot force at any time during the contact is calculated as

$$\mathbf{F}_{ij}^{\text{D}}(t) = \mathbf{F}_{nij}^{\text{D}}(t) + \mathbf{F}_{tij}^{\text{D}}(t) = -\eta_n \mathbf{V}_{nij}(t) - \eta_t \mathbf{V}_{tij}(t), \quad (7.17)$$

For the spring force, at the initiation of the contact, the normal spring force  $\mathbf{F}_{nij}^{\text{S}}$  is equal to  $-k_{nm\ell}\delta_{ij}$  and the tangential component is zero. Unlike for the dashpot forces, a time history of the spring forces is maintained once the contact initiates. At any time during the contact, the normal spring force is given by

$$\mathbf{F}_{nij}^{\text{S}}(t + \Delta t) = \mathbf{F}_{nij}^{\text{S}}(t) - k_n \mathbf{V}_{nij} \Delta t, \quad (7.18)$$

where  $\Delta t$  is the time-step size. The tangential component of the spring force also evolves by a similar expression. For the case of finite Coulomb friction between particles <sup>1</sup>, if the following holds at any time during the contact,

$$|\mathbf{F}_{t_{ij}}| > \mu |\mathbf{F}_{n_{ij}}|, \quad (7.19)$$

then the sliding is assumed to occur and the tangential contact force is given by

$$\mathbf{F}_{t_{ij}} = -\mu |\mathbf{F}_{n_{ij}}| \mathbf{t}_{ij}. \quad (7.20)$$

It is important to note that the  $i^{\text{th}}$  particle in the contact  $i - j$  pair experiences a contact force equal to  $\mathbf{F}_{ij}$  and the  $j^{\text{th}}$  particle, according to Newton's third law of motion, experiences an equal and opposite contact force (i.e.  $-\mathbf{F}_{ij}$ ). Therefore, the total contact force  $\mathbf{F}_c^{(i)}(t)$  at any time on the  $i^{\text{th}}$  particle is given as

$$\mathbf{F}_c^{(i)}(t) = \sum_{\substack{j=1 \\ j \neq i}}^N (\mathbf{F}_{ij}^S(t) + \mathbf{F}_{ij}^D(t)), \quad (7.21)$$

### 7.2.2.2 Relationship between dashpot coefficients and coefficients of restitution

For collisions between particles belonging to the  $m^{\text{th}}$  and  $\ell^{\text{th}}$  solid-phases, the normal dashpot damping coefficient  $\eta_{nml}$  is related to the normal coefficient of restitution  $e_{nml}$  (Silbert et al., 2001) by

$$e_{nml} = \exp\left(-\frac{\eta_{nml} t_{n,m\ell}^{\text{col}}}{2m_{\text{eff}}}\right), \quad (7.22)$$

where  $m_{\text{eff}} = m_m m_\ell / (m_m + m_\ell)$  is the effective mass and  $t_{n,m\ell}^{\text{col}}$  is the collision time between  $m$  and  $\ell$  solid phases. It is given by

$$t_{n,m\ell}^{\text{col}} = \pi \left( \frac{k_{nml}}{m_{\text{eff}}} - \frac{\eta_{nml}^2}{4m_{\text{eff}}^2} \right)^{-1/2}. \quad (7.23)$$

From the above two expressions,  $\eta_{nml}$  is obtained as

$$\eta_{nml} = \frac{2\sqrt{m_{\text{eff}} k_{nml}} |\ln e_{nml}|}{\sqrt{\pi^2 + \ln^2 e_{nml}}}, \quad (7.24)$$

<sup>1</sup>Like for the spring stiffness and dashpot damping coefficients, the friction coefficient  $\mu_{m\ell}$  will also depend on the solid-phases the colliding particles belong to. However, for the sake of clarity, the subscripts are omitted in favor of just  $\mu$ .

and a similar expression can be written for  $\eta_{tml}$ .

The time step  $\Delta t$  is typically taken to be equal to one by fifty of the minimum collision time (i.e.  $\Delta t = \min(t_{col,m\ell}/50)$ ). Specification of spring stiffness coefficients in DEM simulations is problematic. If values close to the real physical values are chosen, then the time step will become very small, prohibiting any large-scale study. Therefore a value of normal spring stiffness coefficient  $\sim 10^5$ , is usually specified. The tangential spring stiffness coefficient is set equal to two-fifths of the normal stiffness coefficient (i.e.,  $k_{tml} = 2/5k_n, \forall m, l$ ). The tangential damping coefficient is generally taken to be half of normal damping coefficient (i.e.,  $\eta_{tml} = 0.5\eta_{nml}, \forall m, l$ ). In gas-particle flows, since the drag force also opposes the particle velocity, a spring stiffness less than that used in pure granular flows can be utilized.

### 7.2.2.3 Estimation of gas-solid momentum transfer term $\mathbf{I}_{gm}$

In this section since we are interested in calculating the momentum interaction term  $\mathbf{I}_{gm}$  between the gas-phase and  $m^{\text{th}}$  solid-phase, the discussion is limited to particles belonging to  $m^{\text{th}}$  solid-phase.

Consider the  $i^{\text{th}}$  particle, belonging to the  $m^{\text{th}}$  solid-phase, that resides in the  $k^{\text{th}}$  computational cell at time  $t$ . The drag force on this particle is represented as

$$\mathbf{F}_d^{(i \in k, m)} = -\nabla P_g(\mathbf{X}^{(i)}) + \frac{\beta_m^{(i \in k)} \mathcal{V}_m}{\varepsilon_{sm}} \left( \mathbf{v}_g(\mathbf{X}^{(i)}) - \mathbf{V}^{(i)} \right), \quad (7.25)$$

where  $P_g(\mathbf{X}^{(i)})$  and  $\mathbf{v}_g(\mathbf{X}^{(i)})$  are the gas-phase mean pressure  $P_g$  and velocity  $\mathbf{v}_g$  fields at the particle location,  $\mathcal{V}_m = \frac{\pi D_m^3}{6}$  is the particle volume, and  $\beta_m^{(i \in k)}$  is the local gas-solid momentum transfer coefficient for particle  $i$  residing in the  $k^{\text{th}}$  cell. An explicit functional form of  $\beta_m^{(i \in k)}$  is not known theoretically and, therefore, different correlations deduced from experimental and numerical studies are used to model this term. Nevertheless, a general parametrization for  $\beta_m^{(i \in k)}$  that subsumes different models can be written as

$$\beta_m^{(i \in k)} = \beta \left( \rho_m, D_m, \left| \mathbf{V}^{(i)} - \mathbf{v}_g(\mathbf{X}^{(i)}) \right|, \rho_g, \mu_g \right). \quad (7.26)$$

The gas-solid momentum transfer term  $\mathbf{I}_{gm}$ , at  $\mathbf{x}_k$ , that enters the gas-phase momentum

conservation equation (Eq. 7.2) is computed as

$$\mathbf{I}_{gm}^k = \frac{1}{\mathcal{V}_k} \sum_{i=1}^{N_m} \mathbf{F}_d^{(i \in k, m)} K(\mathbf{X}_m^{(i)}, \mathbf{x}_k), \quad (7.27)$$

where  $K(\mathbf{X}_m^{(i)}, \mathbf{x}_k)$  is a generic kernel with compact support and determines the influence of the particle force at  $\mathbf{X}_m^{(i)}$  on a grid node located at  $\mathbf{x}_k$ , and  $\mathcal{V}_m$  is the geometric volume of the  $k^{\text{th}}$  grid cell.

In MFIX-CDM, there are two methods available to calculate the above drag force. In the first method, for a particle residing in  $k^{\text{th}}$  cell, rather than computing mean gas-phase velocity at the particle location  $\mathbf{v}_g(\mathbf{X}^{(i)})$ , a cell-centered value of  $\mathbf{v}_g$  is used. Similarly, rather than using velocity of each particle  $\mathbf{V}^{(i)}$ , a local cell averaged velocity of the  $m^{\text{th}}$  solid-phase  $\mathbf{v}_{sm}$  is used. With this simplification, the momentum transfer coefficient for all particles of  $m^{\text{th}}$  solid-phase that reside in cell  $k$  is constant and has the following functional form

$$\beta_m^{(\forall i \in k)} = \beta_m^{(k)} = (\rho_m, D_m, |\mathbf{v}_{sm}(\mathbf{x}_k) - \mathbf{v}_g(\mathbf{x}_k)|, \rho_g, \mu_g), \quad (7.28)$$

where  $\mathbf{x}_k$  is the center of the  $k^{\text{th}}$  cell. Therefore, the drag force on the  $i^{\text{th}}$  particle belonging to solid-phase  $m$  and residing in cell  $k$  is

$$\mathbf{F}_d^{(i \in k, m)} = -\nabla P_g(\mathbf{x}_k) + \frac{\beta_m^{(k)} \mathcal{V}_m}{\varepsilon_{sm}} (\mathbf{v}_g(\mathbf{x}_k) - \mathbf{v}_{sm}(\mathbf{x}_k)). \quad (7.29)$$

Under this approximation of constant drag force on all particles residing in a particular cell, the gas-solid momentum transfer term  $\mathbf{I}_{gm}^k$  is estimated in the  $k^{\text{th}}$  cell as

$$\mathbf{I}_{gm}^k = -\varepsilon_{sm} \nabla P_g(\mathbf{x}_k) + \beta_m^{(k)} (\mathbf{v}_g(\mathbf{x}_k) - \mathbf{v}_{sm}(\mathbf{x}_k)). \quad (7.30)$$

In the second method to calculate gas-solid momentum transfer term, the mean gas-phase velocity is interpolated to the particle location. Using Eq. 7.27, the drag force on each particle is then projected back onto to the Eulerian gas-phase grid. However, in order to avoid the complexities in numerical algorithm that will arise as a result of forward and backward interpolation of the gas-phase pressure field, the pressure drag force term is evaluated at the cell center (resulting in equal pressure drag force on all particles residing in a particular cell).

Therefore, the gas–solid momentum transfer term  $\mathbf{I}_{gm}^k$  is estimated in  $k^{\text{th}}$  cell as

$$\mathbf{I}_{gm}^k = -\varepsilon_{sm} \nabla P_g(\mathbf{x}_k) + \frac{1}{V_k} \sum_{i=1}^{N_m} \frac{\beta_m^{(i \in k)} V_m}{\varepsilon_{sm}} \left( \mathbf{v}_g(\mathbf{X}_m^{(i)}) - \mathbf{V}_m^{(i)} \right) K(\mathbf{X}_m^{(i)}, \mathbf{x}_k). \quad (7.31)$$

### 7.3 Neighbour Search Algorithm

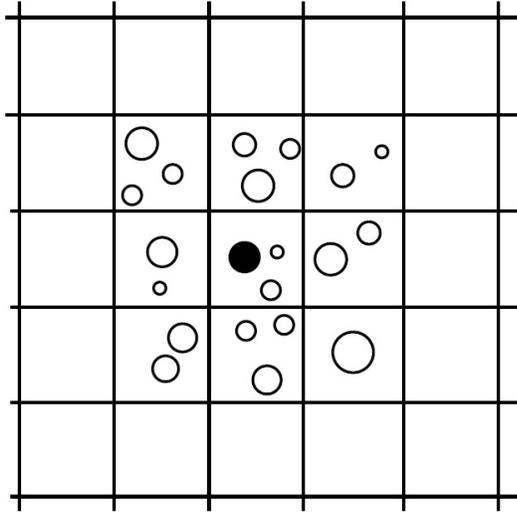


Figure 7.3 2-D Schematic for “cell-linked list” neighbor search algorithm. Hollow and filled circles represent particles of different radii.

One of the most important and time consuming component of any particle-based simulations is the neighbor search algorithm. In MFIX-CDM code, the user has an option to choose between four neighbor search algorithms by specifying an appropriate value for “DES\_NEIGHBOR\_SEARCH” variable in the input file. A value of 1 will use the simplest but the most expensive “ $N^2$ ” search algorithm, where  $N$  is the total number of particles in the domain. Therefore, it should be used either for a small system or for debugging purposes. Input Values of 2 and 3 correspond to the “Quadtree” and “Octree” search algorithms. All the above three methods are grid-free methods. In MFIX-CDM, since the particles are already binned or marked according to the cell they belong to, therefore, it is in one’s advantage to exploit this already existing information. The fourth method, referred to as the “Cell-linked list”

search algorithm and activated by specifying 4 for “DES\_NEIGHBOR\_SEARCH”, is based on this principle. As shown by the 2–D schematic in Fig. 7.3, if the particle of interest is the one represented by the filled circle, then only the particles (shown as hollow circles) that belong to the 9 (27 for the 3-D case) adjacent cells and also to the same cell as the particle of interest are considered as potential neighbors. Any two particles  $i$  and  $j$  that are located at  $\mathbf{X}^{(i)}$  and  $\mathbf{X}^{(j)}$ , and have radii  $R_i$  and  $R_j$ , are considered neighbors if they satisfy the following condition

$$\left| \mathbf{X}^{(i)} - \mathbf{X}^{(j)} \right| < K(R_i + R_j), \quad (7.32)$$

where  $K$  is a user input variable by the name “FACTOR\_RLM” and its default value is equal to 1.2. If “FACTOR\_RLM” is specified as one, then only the particles that are either nearly touching or overlapping will be considered as neighbors. As a result, in order to ensure that the simulation does not miss any possible collisions, the neighbor search algorithm will have to be called at each time step, resulting in high computational expense. A very high value for “FACTOR\_RLM” is also not advisable because then a particle might end up with more neighbors than the array sizes can accommodate, resulting in run–time segmentation errors.

Another important parameter is the frequency at which the neighbor search algorithm is called. In the MFIx–CDM implementation, the neighbor–search algorithm is called every time the code enters the DES modules from the Eulerian solver (irrelevant for pure granular flow simulations). Once in the DES modules, the neighbor search algorithm is called after every, user input, “NEIGHBOR\_SEARCH\_N” number of DES iterations. The default value for “NEIGHBOR\_SEARCH\_N” is equal to 25. Between “NEIGHBOR\_SEARCH\_N” DES iterations, if any particle moves by more than “NEIGHBOR\_SEARCH\_RAD\_RATIO” (user input, default value = 1.0) times its radius, then the neighbor search algorithm is called and the counter for comparing with “NEIGHBOR\_SEARCH\_N” is reset to 0. This second test ensures that if a high value for “NEIGHBOR\_SEARCH\_N” is input, then the simulation doesn’t blow up due to large particle overlaps.

## 7.4 DEM Verification Tests

We wish to perform a series of verification studies for pure granular flows as well as gas-particle flows. The MFIX-CDM code is extremely complex with the interaction between the fluid-solver, particle-solver, collision-algorithms, boundaries etc. In addition, the fluid-solver is on a staggered-grid with scalar quantities solved on the cell centers while the velocities are computed on the cell faces. With all the above complexities, one can only perform very limited verification by visually comparing the code segments to the equations being solved. That is why we are in the process of solving a series of verification tests to probe for the accuracy of each of the units of this complex model.

### 7.4.1 Freely Falling Particle

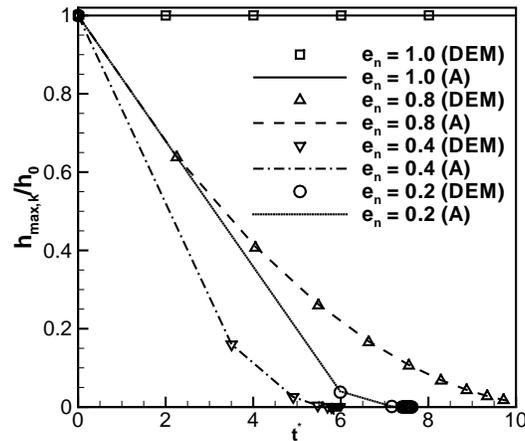


Figure 7.4 Comparison of the evolution of  $h_{\max,k}/h_0$  (ratio of maximum height attained by a freely falling particle under gravity after  $k$  collisions with wall to the initial height) with non-dimensional time,  $t^* = t / \left( e_n \sqrt{\frac{2h_0}{g}} \right)$ , obtained from DEM simulation (denoted DEM) and analytical expression (denoted A) given by Eq. 7.33, for different values of coefficient of restitution  $e_n$ .

In this test case a particle falling freely under gravity bounces upon collision with the

bottom wall. If the particle is dropped from an initial height  $h_0$ , then the maximum height it reaches after its first collision with the wall is  $e_n^2 h_0$ , where  $e_n$  is the coefficient of normal restitution between the particle and the wall. A general expression for the maximum height attained by after  $k$  collisions is

$$h_{\max,k} = h_0 e_n^{2k}. \quad (7.33)$$

Fig. 7.4 shows, for different values of  $e_n$ , the comparison of the evolution of  $h_{\max,n}/h_0$  with non-dimensional time,  $t^* = t / \left( e_n \sqrt{\frac{2h_0}{g}} \right)$ , obtained from DEM simulation (denoted DEM) and the above analytical expression (denoted A). It can be seen that the DEM simulation is able to accurately predict the maximum heights after repeated collisions with the wall.

For any quantity  $Q$ , the relative error  $\epsilon_Q$  between the values predicted by DEM simulation (denoted by  $\{Q\}$ ) and analytically expected values (denoted by  $Q_A$ ) can be defined as

$$\epsilon_Q = \left| \frac{Q_A - \{Q\}}{Q_A} \right|. \quad (7.34)$$

From the above definition, the evolution of the relative error  $\epsilon_h$  in the prediction of  $h_{\max,k}/h_0$  by DEM simulations for different values of coefficient of restitution is shown by Fig. 7.5. It can be seen that the maximum error, which is only about 0.1%, results for purely elastic collisions. For inelastic collisions, the relative error is an order of magnitude less than that for pure elastic collisions.

#### 7.4.2 Ball Slipping on a Rough Surface

In this second verification problem, a spherical ball with finite translational velocity but zero angular velocity is left on a rough surface, also shown by the schematic in Fig. 7.6. As a result of finite slip at the point of contact between the ball and the rough surface, rolling friction will act in the direction shown in Fig. 7.6. This rolling friction will reduce the translational velocity and, at the same time, generate an angular velocity until there is zero slip at the point of contact, i.e.  $v = \omega R$ . After the zero slip condition is reached, rolling friction will cease to act and the solid ball keep on moving with fixed translational and angular velocities.

From the force balance shown in the free body diagram, the normal contact force  $F_n = W = mg$ , where  $W$  and  $m$  are, respectively, the weight and mass of the spherical ball, and

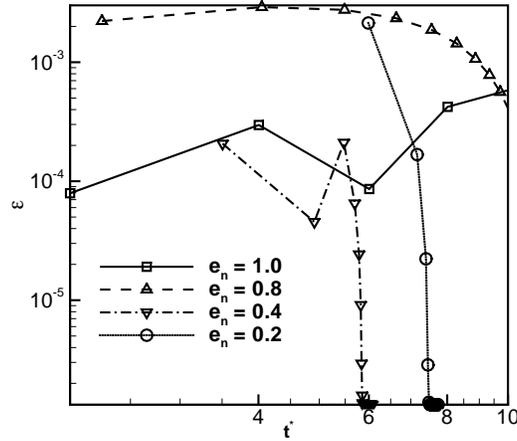


Figure 7.5 Evolution of the relative error  $\epsilon_h$  (Eq. 7.34) in estimation of  $h_{\max,k}/h_0$  by the DEM simulations for different values of coefficient of restitution.

$g$  is the acceleration due to gravity. The tangential contact force  $F_t$ , which is the force due to rolling friction, is equal to  $\mu mg$ . Therefore, the evolution equations for translational and angular velocities become

$$\frac{dv_x}{dt} = -\mu g, \quad (7.35)$$

and

$$\frac{d\omega}{dt} = \frac{\mu mg R}{I}, \quad (7.36)$$

where  $I = 2/5 m R^2$  is the moment of inertia of the spherical ball. The above equations can be intergrated with the initial conditions  $\{v_x, \omega\}_{t=0} = \{v_0, 0\}$ , where  $v_0$  is the initial translational velocity of the ball. Since the evolution equations for  $v_x$  and  $\omega$  are known, the time  $t_s$  at which slipping ends (i.e.  $v_x = \omega R$ ), or rolling friction ceases to act, can be calculated analytically.

This time  $t_s$  is

$$t_s = \frac{2v_0}{7\mu g}. \quad (7.37)$$

The non-dimensional translational and angular velocities at  $t_s$  are

$$\{v'_x, \omega'\}_{t=t_s} = \left\{ \frac{v_x}{v_0}, \frac{\omega R}{v_0} \right\}_{t=t_s} = \left\{ \frac{5}{7}, \frac{5}{7} \right\} \quad (7.38)$$



Figure 7.6 Schematic of the second verification problem. A spherical ball with finite translational velocity and zero angular velocity is placed on a rough surface. Forces acting on the ball is shown by the free body diagram on the right.

Fig. 7.7 shows the comparison of  $t' = \mu g t_s / v_0$  (left axis), and  $\{v'_x, \omega'\}_{t=t_s}$  (right axis) obtained from DEM simulation with the analytic values for different values of coefficient of friction. The relative error, not shown, is always less than 0.1%.

## 7.5 Future Work

In addition to the two verification tests discussed above, a thorough validation of the MFIX-CDM code is underway. Followig validation and verification tests are planned as the future work:

- Elastic particle bouncing of the wall centers at  $45^\circ$  in a square enclosure with no gravity
- Two elastic particles bouncing of each other an the walls in the center of a square enclosure with no gravity
- Inelastic particle bouncing of the bottom wall under gravity
- Frictional particle sliding on bottom wall under gravity with initial velocity
- Particle motion in one dimension with all the DEM forces (Chen et al., 2007).
- Particle motion in vortex flow
- Particle motion in fluid flow at an angle

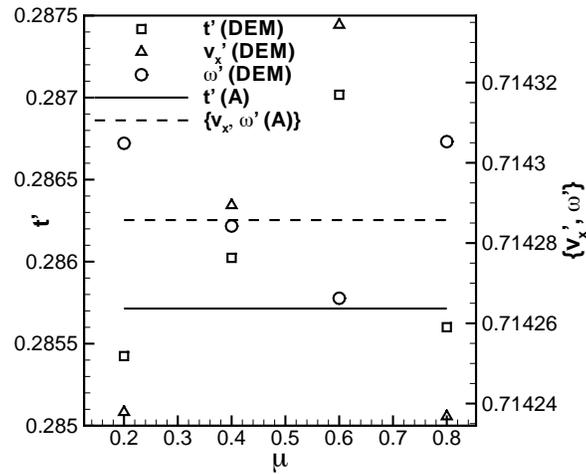


Figure 7.7 Comparison of  $t' = \frac{\mu g t_s}{v_0}$  (left axis), and  $\{v'_x, \omega'\}_{t=t_s}$  (right axis) obtained from DEM simulation with the expected values for different values of coefficient of friction.

- Particles in sphere stretching/unstretching flow

All the above tests will be subjected to the following computational tests to ensure sanity

- Invariance to coordinate changes
- Grid convergence

A fully verified/validated MFIx-CDM code will then be applied to typical validation cases such as bubbling bed with a jet, granular discharge of particles with and without gas.

## CHAPTER 8. DISCUSSION AND CONCLUSIONS

The principal objective of improving the predictive capabilities of two-phase simulations that solve for averaged equations, such as LE, EE, and QMOM, has been achieved by improving the accuracy of the fluid-particle interaction term. The accuracy of the fluid-particle interaction term has been improved on two fronts. First, the accuracy has been improved by proposing new correlations for the fluid-particle interaction terms, such as interphase momentum transfer (in the form of a drag law for monodisperse particles), and interphase heat and mass transfer (in the form of a Nusselt/Sherwood number law for monodisperse particles) using the ‘true’ DNS approach. Second, the numerical accuracy of estimating fluid-particle interaction term is investigated and then improved in LE simulations. Special emphasis has been laid on LE simulations as they are a useful tool in the verification and development of other modeling and numerically wise intensive simulation types, such as EE and QMOM. The contributions, conclusions and implications of each sub-study are discussed below in separate sections.

### 8.1 Conclusions from modeling study of fluid-particle interaction term

A ‘true’ DNS approach based on the discrete time immersed boundary method, first proposed by Yusof (1996), is extended to solve for flow and scalar transport past homogeneous assemblies of fixed particles. Some of the major contributions, conclusions, and implications of this study are:

1. IBM was successfully extended to solve for flow past homogeneous assemblies of particles.

An important correction to the original immersed boundary scheme proposed by Yusof (1996) is made to ensure non-contamination of fluid velocity, pressure, and scalar fields

by IB forcing.

2. A theoretical connection of quantities computed from DNS was made with the unclosed terms appearing in the averaged equations resulting from both Eulerian and Lagrangian statistical representations of the solid-phase.
3. A first of its kind comprehensive numerical convergence study was performed for ‘true’ DNS of gas-solids flow that highlights the need for high numerical resolutions with increasing Reynolds number of solid volume fraction.
4. Hydrodynamic IBM solver was extensively validated by comparing with the past analytical, experimental, and numerical studies. IBM is shown to be a robust tool for ‘true’ DNS of gas-solids flow for arbitrary values of solid volume fraction and Reynolds number, provided the stability criteria are met. Furthermore, it is shown that IBM does not suffer from the limitations of LBM, which is another ‘true’ DNS approach used in the past for proposing drag laws for homogeneous assemblies of monodisperse and bidisperse particles.
5. The hydrodynamic solver was successfully extended to solve for scalar and heat transport past homogeneous assemblies of monodisperse particles. This is the first of its kind DNS study of scalar and heat transport in gas-solids flow.
6. The average Nusselt number obtained from scalar IBM simulations is found to be in reasonable agreement with the experimental findings of Gunn and Desouza (1974) and Pfeffer and Happel (1964) in the Stokes flow regime, and with the experimental findings of Turner and Otten (1973), and Denton (1951) for low to high Reynolds numbers ( $Re < 300$ ).
7. The average Nusselt in the Stokes regime, however, is found to be in major disagreement with the widely used, semi-analytical, heat and mass transfer correlation of Gunn (Gunn, 1978), with difference as high as 300% for the highest solid volume fraction of 0.5 considered in this study. Such high differences between scalar IBM results and Gunn’s

correlation is primarily attributed to the neglect of axial diffusion in the Stokes flow regime by Gunn.

8. A new heat and mass transfer correlation as function of solid volume fraction and Reynolds number is proposed for a particular Prandtl/Sherwood number value of 0.7. The new correlation is recommended to be used in averaged equation solvers (such as LE, EE, and QMOM) for studies related to heat and mass transfer in fixed and fluidized gas–solids flows.
9. A three–step “random configurations initialization” algorithm has been developed for initializing random arrays up to very dense solid volume fractions ( $\approx 0.52$ ). The three–step algorithm has the additional ability to generate random arrays having the same volume fraction and number density, but differing in hard–core distance  $h_c$  (defined as the minimum distance between the centers of any two particles). It is shown that much higher values for hard–core distance are accessible through the three–step algorithm than those possible from stochastic methods, such as the Matèrn point–process (Stoyan and Stoyan, 1995). The three–step algorithm is a useful tool to study the effects of inter–particle distance on mean quantities, such as the drag and average Nusselt number.

It is worthwhile to note that the implications of accurate closures/models for fluid–particle interaction terms are not limited to just the averaged equation solvers, such LE, EE, and QMOM. The implications of this study apply to any two–phase flow simulation types that rely on closures for the fluid–particle interaction terms. These include simulations of averaged equations (LE, EE, and QMOM) and also the simulations of instantaneous equations (point–particle DNS, LES).

As a matter of fact, availability of ‘true’ DNS data will greatly aid in the development of LES simulation method for two–phase flows. This is because the filtering operation for two–phase flows, just like the averaging procedure, results in additional residual terms for different source terms, such as, interphase momentum transfer or energy transfer, which are generally referred to as subgrid source terms (in order to differentiate them from subgrid

terms, generally used in the context of single-phase flows). In LES of two-phase flows, the subgrid source terms have traditionally been neglected. For example, Wang and Squires (1996); Uijtewaal and Oliemans (1996) did not consider the two-way coupling and subgrid effects in source terms. Boivin et al. (2000); Yamamoto et al. (2001) considered the two-way coupling but the subgrid effects in source terms were not accounted for. From the fully resolved fields (for both flow and scalar) obtained from ‘true’ DNS, very accurate models for the subgrid source terms can now be proposed. This is similar to the approach adopted in development of reliable LES methods in single-phase flows, where well established models for subgrid terms have been obtained from accurate DNS of isotropic turbulence (Kraichnan, 1976; Chasnov, 1991) and fully developed turbulent channel flows (Moin and Kim, 1982; Piomelli, 1993). In the absence of ‘true’ DNS, point-particle DNS (‘pp’ DNS) has been used in the past to develop models for the subgrid source terms (Okong’o and Bellan, 2004). However, the assumption of point-particles in ‘pp’ DNS precludes boundary layer effects, and, as a result, increased dissipation around particles that is essential for accurately quantifying turbulence attenuation by the particles (Hwang and Eaton, 2006) is not captured in ‘pp’ DNS. The ‘true’ DNS approach, on the other hand, being a first-principles approach does not suffer from the same limitations, and it will greatly contribute to the future development of accurate models for subgrid terms needed for quantitatively accurate LES of two-phase flows.

## 8.2 Conclusions from study of numerical accuracy and convergence characteristics of LE simulations

Among all the simulations types for averaged equations in two-phase flows, LE simulations are modeling wise the least restrictive on the range of physical regimes that they can legitimately represent. This is because, only fluid-particle interactions, in the form of interphase momentum transfer and interphase heat and mass transfer terms, need to be modeled in LE simulations. On the other hand, in EE simulations, in addition to the fluid-particle interaction term, modeling of particle-particle interaction term is also required. Therefore, LE simulations are typically used to test the various models for particle-particle interaction term by compar-

ing results from LE and EE simulations. Furthermore, LE simulations are generally used as a benchmark tool in the development of new simulation techniques for two-phase flows, such as the recent quadrature method of moments (QMOM) proposed by Fox (2008). Therefore, the need for accurate LE simulations that can reliably be used for quantitative comparisons with other simulation types cannot be over emphasized.

Accurate numerical estimation of the interphase transfer terms from finite number of particle used to represent the dispersed phase in LE simulations is most crucial for accurate LE simulations. With regard to the numerical accuracy of LE simulations, some of the major contributions, conclusions, and implications of this study are:

1. An error model, originally proposed by Xu and Pope (1999), is extended to estimation of mean interphase momentum transfer term in LE simulations. The error model decomposes the error in estimating mean momentum transfer term as the sum total of statistical, bias, and discretization errors. The error model is used to characterize the convergence properties of four interpolation schemes for mean interphase momentum transfer term used in LE simulations in a series of static test problems. 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 motivated the development of more accurate estimator, termed as improved estimator, that 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.
2. It is shown through a simulation of two-way coupled particle-laden lid-driven cavity flow that traditional LE simulations, that use real or computational particles with constant statistical weight, fail to yield numerically converged solutions when solved on progressively refined grids. 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. It is concluded that the combination of ILE with the improved estimator will yield numerically convergent and accurate results for two-phase flows that can reliably be used for quantitative comparison with other simulations types and with experiments as well.

It is worth noting that although the above work has been motivated around LE simulations, but the above conclusions apply equally well to any particle based two-phase simulation techniques. For example, DNS and LES of two-phase flows that employ point source representation for the dispersed phase. In single-phase flows, Large-Eddy simulations (LES) have proved to be a useful and reliable alternative to DNS simulations. Since the smallest scales are not resolved in LES simulations, the computational cost requirement is significantly reduced. The filtering operation in LES methodology results in residual terms accounting for the unresolved small scale terms. These residual terms are accounted in LES simulations through various subgrid models available in the literature. They find that all the subgrid models overestimate the corresponding DNS filtered data, and also observe a complex dependence of prospective models on the filter width. Developing accurate subgrid models for various source terms in two-phase flows is an active area of research.

In addition to the accurate closures for fluid-particle interactions, accurate closures are required for particle-particle interactions in EE simulations. In the next section, there is a brief discussion on the closures for particle-particle interaction term used in EE simulations and the resulting restrictions on physical systems that the current EE simulations can legitimately solve for. The recent QMOM simulation technique, being developed by Dr. Rodney Fox and his students in chemical engineering at Iowa state university, that relaxes the limitations posed by KTGF closures is also discussed. The numerically converged and accurate ILE simulation method developed in this work is used as a benchmark in the comparison of EE and QMOM simulations.

### 8.3 Implications of KTGF closures for particle–particle interaction term in EE simulations

The closures or constitutive relations for the particle–particle interactions required in EE simulations have been obtained by studying particle dynamics in the absence of gas phase, resulting in the so called granular flows. For dilute to moderately dense flows, closures for particle–particle interactions are obtained from the extension of classical kinetic theory for molecular gases (Chapman and Cowling, 1953) to granular gases (i.e., KTGF). Since its inception by Savage and Jeffrey (1981), KTGF over the last three decades has undergone tremendous theoretical improvements that we choose not to dwell on in this brief overview. However, some inherent assumptions made in KTGF limits it to physical systems that it can correctly describe. Some of the limitations in the context of two–phase flows are discussed below.

KTGF assumes binary and instantaneous collisions, therefore, it can only describe the transport in collision dominated or *rapid flow regimes*. For dense systems, also referred to as *plastic flow regime*, the dominant momentum transport between particles is due to enduring contacts. Therefore, in the plastic regime, the particle–particle interaction term is closed by adopting theories from the study of soil mechanics (Tuzun et al., 1982; Jackson, 1983), where the stresses in the particle phase are assumed to arise because of particle friction, and are described by phenomenological models rather than mechanistic models as in the case of rapid flow regime described using KTGF.

On the other extreme of the plastic regime is the case of very dilute flows where again the KTGF closures are not suitable due to the strong assumption of collision–dominated flows (which are further assumed to be nearly at equilibrium, resulting in the assumption of Maxwellian particle velocity distribution) made in KTGF. The assumptions of collision–dominated flows along with small departure from equilibrium limits the applicability of KTGF closures to low Knudsen ( $Kn < 0.1$ ) numbers. For high Knudsen number systems (very dilute systems or high Mach number systems), where rarefaction effects are not negligible, it is now known that the particle velocity distribution is not Maxwellian (see Campbell (1990) and the references therein). Although KTGF studies (Garzó and Dufty, 2002; Iddir et al., 2005; Garzo

et al., 2007b) that assume non-Maxwellian velocity distribution have been completed recently, the theory is still far from being successfully applied to arbitrary values of Knudsen number. As a result of the above assumptions, KTGF closures when applied to high Knudsen number systems will lead to erroneous predictions of the physical phenomena.

Since the KTGF closures are obtained for granular flows, only conservative body forces, such as the gravity force, are assumed to act on the particles. However, the presence of a carrier fluid in two-phase flows introduces additional and more complicated physics, the most important of them being the non-conservative drag force that acts on the particles. In two-phase flows, the Stokes number  $St$ , defined as the ratio of particle response time to fluid-phase characteristic time scale, is an important parameter. For very low Stokes numbers ( $St \ll 1$ ), particles follow the fluid streamlines very closely. However, for high Stokes number flows, it is commonly observed that particles preferentially concentrate (Squires and Eaton, 1991) in high-strain regions of the carrier phase flow field. Recently, using an impinging particle jets problem, Desjardins et al. (2008) showed that EE models, relying on KTGF closures, are unable to correctly capture the so called particle trajectory crossing phenomena, wherein a particle can readily cross the plane without the risk of collisions with other particles. Since KTGF closures do not include Stokes number effects, any existing particle-particle interaction model will not be able to capture particle trajectory crossing effects.

Due to severe restrictions posed by using KTGF closures, Fox (2008) proposes solving the discrete form of the kinetic equation through the use of quadrature method of moments (QMOM). QMOM is a first-principles approach to solving the kinetic equation for arbitrary Knudsen number (including the Euler limit), and hence removes the limitation on very small range of Knudsen number that can be legitimately solved using KTGF closures. In addition to the relaxation on Knudsen number range, QMOM does not assume any base state (like the assumption of homogeneous cooling state in KTGF studies, implying a Maxwellian velocity distribution) and, therefore, in principle, it is capable of capturing any velocity distributions. Since the method is developed in the context of two-phase flows, non-conservative body forces, such as the drag force (versus only the conservative gravity force assumed in KTGF), is assumed

to act on the particles. The effect of the drag force is directly accounted for in the evolution equation of one particle distribution function (i.e., the kinetic equation) itself. As a result, opposed to the KTGF based simulations (like EE), QMOM is naturally capable of capturing the additional physics commonly observed in multi-phase flows due to finite Stokes number effects (such as particle clustering and particle trajectory crossings). A quadrature-based third-order moment closure is derived by Fox (2008) which is applicable to gas-particle flows at any Knudsen number. Currently, the method is limited to Boltzmann like kinetic equation<sup>1</sup>, thus, limiting it to dilute volume fractions. By considering an Enskog like kinetic equation, although numerically challenging, the method can in principle be extended to moderately dense volume fractions.

From the one-way coupled test problems, used as proof of concept in Fox (2008), the method has been successfully extended to two-way coupled particle-laden flows in Passalacqua et al. (2009). In Passalacqua et al. (2009), as an example application, simulations of a two-way coupled particle-laden vertical channel flow are carried out. For particle Stokes number nearly equal to one, instabilities leading to the formation of structures and initiating particle segregation process are observed from QMOM simulations. Using the ILE simulation method developed in this work, LE simulation of the same system also predict the formation of similar structures. However, EE simulations, relying on KTGF closures, are unable to predict formation of such structures. This is attributed to the locally high Knudsen numbers observed in this flow which are well beyond the rather small range of Knudsen number ( $Kn < 0.1$ ) where KTGF closures are valid.

The comparison of the modeling requirements and computational cost between LE and EE simulation approaches results in a paradox wherein the modeling wise less intensive LE approach is computationally prohibitive to solve device-scale problems, and, on the other hand, computationally viable EE approach requires extra modeling (which severely limits its applicability to all values of solid volume fraction, Knudsen and Stokes numbers) to describe particle-particle interaction term  $I_{pp}$ . QMOM, on successful extension to dense flows, will

<sup>1</sup>A closed form for the collision term is assumed using the Bhatnagar-Gross-Krook (BGK) approximation (Bhatnagar et al., 1954).

prove to be an excellent mean to simulate two-phase flows as it is computationally as viable as the EE approach to solve device-scale problems, has fewer modeling assumptions for the particle-particle interaction term than in EE approach, and does not suffer from statistical errors as does the LE approach.

## APPENDIX A. RANDOM CONFIGURATIONS INITIALIZATION

For low volume fractions, a random configuration can be easily generated by releasing the particles (originally seeded in a lattice arrangement) with a Gaussian velocity distribution and letting them collide elastically. For low volume fractions, a hard-sphere (Allen and Tildesley, 1989) can be used to collide the particles. Theoretically, using the FCC lattice arrangement, close packing limit of the random arrays can be accessed. However, the box length to particle diameter  $L/D$  in our simulations is as small as 4 and the maximum value considered is 10. Therefore, for such relatively small systems, due to the deletion of particles at the edges, it is difficult to fit in all the particles inside the box. In order to avoid this difficulty, we use a three-step procedure to generate random particle configurations in our study.

In our study, for a given volume fraction, we are interested in studying the effect of inter-particle distance on drag force and Nusselt number. The inter-particle distance is best quantified by the hard-core distance  $h_c$  which is the minimum distance between the centers of any two particles in a suspension. For dilute volume fractions, Matèrn point-process (Stoyan and Stoyan, 1995) is a convenient mean of generating random distributions with a specified hard-core distance. The maximum hard-core distance possible with Matèrn point-process is

$$h_{c,\max}^{\text{Mat}} = \frac{D}{2} \varepsilon_s^{-1/3}. \quad (\text{A.1})$$

For volume fractions equal to 0.01 and 0.1, the above expression yields maximum hard-core distances of 2.32 and 1.077, respectively. However, we are interested in even higher values of hard-core distances than those possible by Matèrn point-process. The three-step algorithm is able to generate configurations with even higher hard-core distances as described below.

In order to generate distributions with different hard-core distances, the three-step algo-

rithm described below starts with an increased diameter  $\widehat{D}$  given as

$$\widehat{D} = D \left[ 1 + \frac{h_m}{D_m} + \frac{X}{100} \right], \quad (\text{A.2})$$

where  $D$  is the actual desired particle diameter,  $h_m$  is the minimum number of grid cells between the surfaces of any two particles, and  $D_m = D/\Delta x$  is the number of grid cells across a particle diameter. Since we use the soft-sphere collision model, although small but there will be positive overlaps between the particles. In order to ensure non-overlapping particles for the case of  $h_m = 0$ , a safety factor of  $X\%$  has been added to the above expression for  $\widehat{D}$ . If the spring stiffness parameters are appropriately chosen in the soft-sphere model, then overlaps are typically below 1%. Therefore, we have used a safety factor of 1% (i.e.,  $X = 1$ ) for generating all random distributions in our study. The final particle configuration has a hard-core distance  $h_c = \widehat{D}$  and different hard-core distances can be achieved by varying the value of minimum separation between the particle surfaces  $h_m$ . In order to ensure that the number of particles initialized remain the same as those implied by the desired diameter  $D$  and volume fraction  $\varepsilon_s$ , the volume fraction corresponding to  $\widehat{D}$  is also increased by

$$\widehat{\varepsilon}_s = \varepsilon_s \left( \frac{\widehat{D}}{D} \right)^3 = \varepsilon_s \left( 1 + \frac{h_m}{D_m} + \frac{X}{100} \right)^3. \quad (\text{A.3})$$

At the end of step 3 of the below algorithm, diameter of the particles is reverted back to the desired diameter  $D$  by using the above expression (Eq. A.2).

With the help of a representative case of  $L/D = 8$  (such that  $D=1$ ),  $\varepsilon_s = 0.4$ , and  $h_m = 0$ , each step of the initialization algorithm is outlined below for:

1. *Lattice arrangement:* In the first step, the particles are arranged in a simple cubic lattice arrangement such that they are well inside the box dimensions in the  $xz$ - plane. In the  $y$ - direction, however, the particles are allowed to extend beyond the box size  $\mathcal{L}_y$ . The minimum and maximum value of the particle centers in the  $y$ - direction is calculated as  $Y_{\min}$  and  $Y_{\max}$ . Fig. A.1(a) shows the particle configuration obtained for the representative case.
2. *Shrinkage and mapping:* If  $Y_{\max} - Y_{\min} > \mathcal{L}_y - \widehat{D}$  (as is clearly evident from Fig. A.1(a) for the representative case), that implies an overflow from the above lattice arrangement

procedure, otherwise the procedure advances to step 3. In order to bring in all the particles within  $\mathcal{L}_y$  length, particles are squeezed in from both the top and bottom. This is achieved by applying a symmetric, exponentially decaying gravity force as

$$\mathbf{F}_{\text{shrink}} = mg\widehat{\mathbf{Y}}'_p \left[ 1 - \exp\left(-4Y_p'^2/\mathcal{L}_y'^2\right) \right], \quad (\text{A.4})$$

where  $\mathbf{Y}'_p = \mathbf{Y}_p - \mathcal{L}'_y/2$ , such that  $\mathbf{Y}_p$  is the  $y$ - co-ordinate of the particle center,  $\widehat{\mathbf{Y}}'_p = \mathbf{Y}'_p/Y'_p$  is the unit vector, and  $\mathcal{L}'_y = Y_{\text{max}} + 0.5\widehat{D}$  is the  $y$ - length of the extended box from step 1. Particles are released with a Gaussian velocity distribution and a soft-sphere (Cundall and Strack, 1978) model is used to model the collisions between the particles. The above symmetric gravity force pushes the particles in from both directions. So that the particles at the outer edges along the  $y$ - direction do not drift even further away on colliding with the neighboring particles, a very low value of coefficient of restitution (such as 0.2) is used in this step. The above procedure is performed for some arbitrary stop time  $t_{\text{stop1}} = 20\sqrt{2\mathcal{L}'_y/g}$ . If the condition  $Y_{\text{max}} - Y_{\text{min}} < \mathcal{L}_y - \widehat{D}$  is achieved within this time, then the procedure advances to step 3 outlined below. However, if this condition is not met, then the particles are reinitialized with a new Gaussian velocity distribution and this step is repeated until all the particles fall within length  $\mathcal{L}_y$  in the  $y$ - direction . At the end of this step, particle center positions in  $y$ - direction are linearly mapped from  $(0, \mathcal{L}'_y)$  to  $(0, \mathcal{L}_y)$ . Fig. A.1(b) shows the particle configuration after shrinking and mapping.

3. *Homogeneous particle distributions:* We are interested in homogeneous particle position distributions. For low volume fractions, lattice seeding will suffice and step 2 will not be needed. For dense systems, step 2 will generate a particle position distribution with strong spatial behavior in  $y$ - direction. Therefore, regardless of step 2 being called or not, particle position distribution at the beginning of this step is not homogeneous. In order to generate a homogeneous distribution, the particles are released with a Gaussian velocity distribution (with variance  $\sigma^2$ ) such that they collide (using soft-sphere model) elastically. This procedure is carried out for a stopping time  $t_{\text{stop2}}$ , based on the mean

free path and particle velocity variance, equal to

$$t_{\text{stop}2} = 10 \frac{\lambda_{\text{mfp}}}{\sigma}, \quad (\text{A.5})$$

where  $\lambda_{\text{mfp}} = (\mathcal{V}/N_s)^{1/3}$  is the mean free path based on the number density. Fig. A.1(c) shows the particle position configuration for the representative case obtained after elastically colliding the particles for  $t_{\text{stop}2}$ . At this point, the diameter of the particles is reverted back to the desired diameter  $D$  by using expression (Eq. A.2) so that we get the desired volume fraction  $\varepsilon_s$  and hard-core distance  $h_c$ .

The above procedure guarantees truly random and homogeneous particle position distributions. With the above shrinking technique in step 2, we are able to shrink solid volume fractions in the range 0.5 – 0.55. Therefore if the maximum volume fraction that can be shrunk is  $\varepsilon_{s,\text{max}}$ , then the maximum hard-core distance (from Eq. A.3 and noting that  $h_c = \widehat{D}$ ) that we can generate for a given volume fraction  $\varepsilon_s$  is

$$h_{c,\text{max}} = D \left( \frac{\varepsilon_{s,\text{max}}}{\varepsilon_s} \right)^{1/3}. \quad (\text{A.6})$$

Comparing the above expression with the maximum possible hard-core distance by Matèrn process  $h_{c,\text{max}}^{\text{Mat}}$  (Eq. A.1), reveals that  $h_{c,\text{max}} = 2\varepsilon_{s,\text{max}}^{1/3} h_{c,\text{max}}^{\text{Mat}}$ . If  $\varepsilon_{s,\text{max}} = 0.53$ , then for desired volume fractions  $\varepsilon_s$  equal to 0.01 and 0.1, the maximum possible hard-core distances  $h_{c,\text{max}}$  are equal to  $3.75D$  and  $1.74D$ , which are nearly 1.5 times then those possible by the Matèrn point-process. With the use of more sophisticated shrinking techniques (like the one implemented in LAMMPS (Plimpton, 1995)), it is possible to shrink volume fractions even upto the close packing limit; thus, allowing for even higher hard-core distances than those possible by the current algorithm.

For homogeneous suspensions, particle pair-correlation function  $g(r)$  is a good measure of the suspensions' micro-state and it should peak at the hard-core distance  $h_c$ . Fig. A.2(a) shows the particle pair-correlation function (obtained by averaging over 100 independent realizations) for different values of minimum grid cells between particle surfaces  $h_m$  for solid volume fraction  $\varepsilon_s$  equal to 0.01, particle diameter resolution  $D_m = 10$ , and box-length to particle diameter ratio  $L/D = 15$ . Fig. A.2(b) shows the same comparison for  $\varepsilon_s = 0.1$ ,  $D_m = 20$ , and  $L/D = 7.5$ . In

both the figures it can be seen that the above three-step algorithm results in pair correlation functions that are typical of the homogeneous suspensions, i.e., peaking close to the hard-core (given by Eq. A.2 for different values of  $h_m$ ) distances and then dying off. Therefore, the above three-step algorithm provides a robust way of generating random particle configurations and also allows for studying the second-order effects due to different particle pair-correlation functions over a wider range of hard-core distances than those possible by Matèrn point-process.

Volume fractions higher than 0.53 can be generated using Zinchenko's (Zinchenko, 1994) algorithm. In our study, since the maximum volume fraction considered is 0.5, the above three-step algorithm is used to generate all the random configurations.

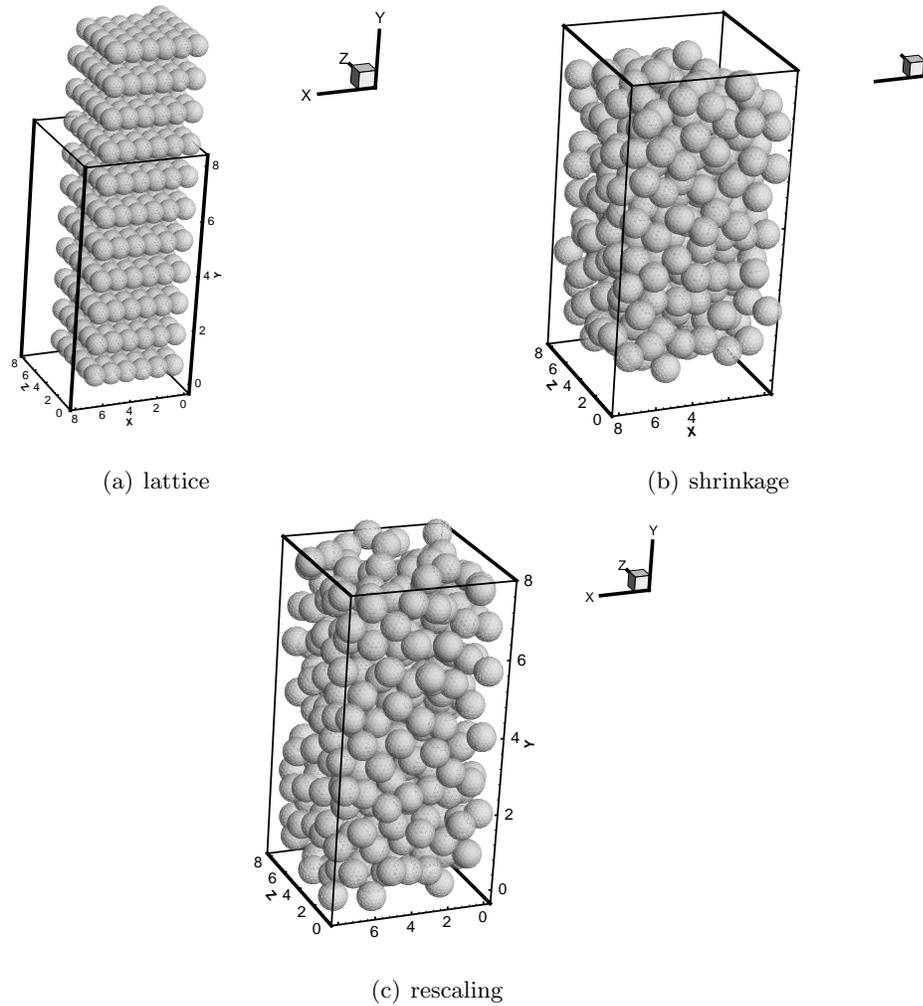


Figure A.1 Particle position configurations obtained for the representative case ( $L/D = 8$ ,  $D = 1$ ,  $\varepsilon_s = 0.4$ ) after (a) lattice distribution (step 1), (b) shrinkage and mapping (step 2), and (c) elastic collisions (step 3).

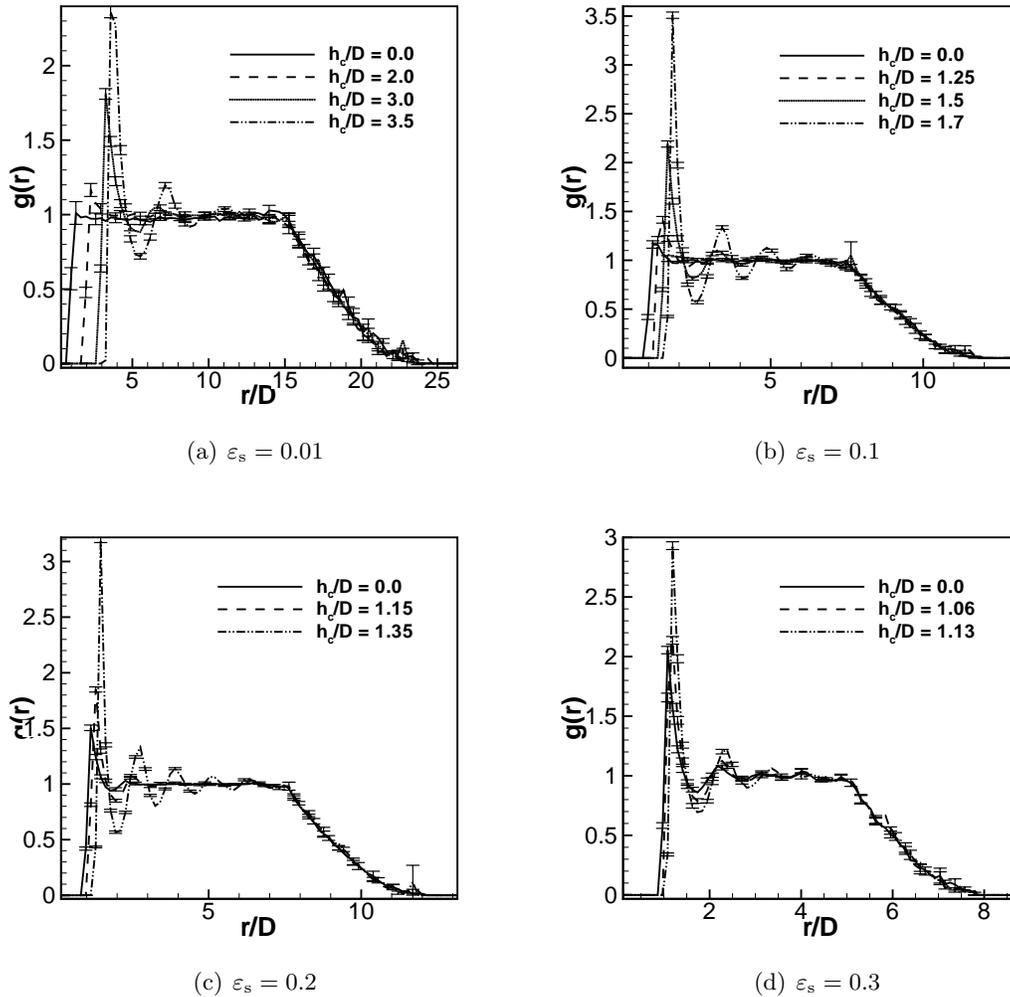


Figure A.2 Pair correlation function  $g(r)$  for different values of  $h_m$  (minimum grid cells between particle surfaces) obtained by averaging over 100 multiple realizations. In (a)  $\varepsilon_s = 0.01$ ,  $D_m = 10$ , and  $L/D = 15$ , and in (b)  $\varepsilon_s = 0.1$ ,  $D_m = 20$ , and  $L/D = 7.5$ . A safety factor of 1% (i.e.,  $X = 1$  in Eq. A.2) has been used.

## APPENDIX B. DETAILS OF INTERPOLATION SCHEMES

For the simplest PIC method Crowe (1982), the estimate for mean interphase momentum transfer term at grid node  $m$  enclosing volume  $V_m$  is given by

$$\{\mathbf{F}_m^{fp}\} = \frac{1}{V_m} \sum_{k=1}^{N_v} \mathbf{f}^k n_p^k, \quad (\text{B.1})$$

where  $N_v$  is the number of particles contained in volume  $V_m$ , and  $n_p^k$  is the number of physical particles represented by the  $k^{\text{th}}$  computational particle.

In order to avoid complicated expressions arising from expressing the interpolation schemes in 3-D, only 1-D formulations are given with the reference to 1-D grid shown in Fig. B.1. These expressions can be readily extended to three dimensions. In B.1,  $m$  is the grid point index and  $c$  is the grid cell index.

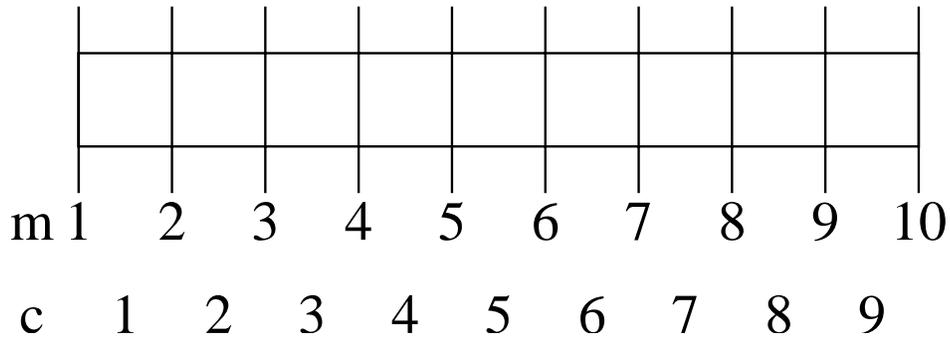


Figure B.1 1-D grid showing grid nodes and cells.

LPI-2, LPI-4 and PCA can be expressed in a very general way for both forward interpolation and backward estimation. The formulation for TSE is slightly different and will be discussed separately. Fluid velocity  $\{U^f(X^k)\}_M$  at the  $k^{\text{th}}$  particle's location, which belongs to  $c^{\text{th}}$

cell, is given by the summation of product of fluid velocities at grid nodes  $U^f(x_m)$  and basis functions  $b_l^x$  or

$$\left\{U^f(X^k)\right\}_M = \sum_{m=c-\mathcal{O}/2+1}^{c+\mathcal{O}/2} U^f(x_m) b_l^x(\xi_l^k), \quad (\text{B.2})$$

where  $\mathcal{O}$  is the order of the scheme which is two for LPI-2 and four for both LPI-4 and PCA,  $l = m - c + \mathcal{O}/2$ , and  $\xi_l^k$  is the elemental coordinate that is defined for each scheme in the following subsections. The convention followed in the above equation numbers the basis functions from left to right. For example, if a particle is located in 5<sup>th</sup> cell (i.e.  $c = 5$ ), then the fourth order LPI-4 interpolation scheme will yield four non-zero basis functions  $b_1$  through  $b_4$ , and the fluid velocity at particle location will have contributions from grid nodes 4 through 7 or  $m = 4, 7$  in the above summation. Based on the convention followed, the basis function that adds the contribution of fluid velocity at 4<sup>th</sup> grid node is numbered 1 while the one for 7<sup>th</sup> grid node is numbered 4.

Similarly, a general expression for the mean interphase momentum transfer term  $\left\{F_{x,m}^{fp}\right\}$  at  $m^{\text{th}}$  grid node is given by

$$\left\{F_{x,m}^{fp}\right\} = \frac{1}{V_m} \sum_{c=m-\mathcal{O}/2}^{m+\mathcal{O}/2-1} \sum_{k=1}^{N^c} f_x^k n_p^k b_l^x(\xi_l^k), \quad (\text{B.3})$$

where  $N^c$  is the number of computational particles in  $c^{\text{th}}$  cell, and all the other quantities have the same meaning as before. To clarify the above equation, consider the case of a fourth ordered scheme. From B.1, the estimate for  $\left\{F_{7,x}^{fp}\right\}$  will include the contribution from particles located in cells 5, 6, 7 and 8. The above method for estimating the mean interphase momentum transfer has been widely used in simulation of particle-laden flows. For example, Boivin et al. (1998) uses the second order linear interpolation (LPI-2), and Sundaram and Collins (1996) uses the fourth order Lagrange polynomial interpolation (LPI-4) scheme.

The next three subsections that follow defines the basis functions for LPI-4, LPI-2, and PCA. In addition, the last subsection completely describes the two stage estimation (TSE) algorithm which so far has not been explained.

### B.1 Linear Interpolation (LPI-2)

LPI-2 is a second-order scheme. For a point  $x$  that lies in the interval  $[x_m, x_{m+1}]$ , it has two linear basis functions

$$\begin{aligned} b_1^x &= 1 - \xi \\ b_2^x &= \xi, \end{aligned} \quad (\text{B.4})$$

where  $\xi$  is the elemental coordinate defined as

$$\xi = \xi_1 = \xi_2 = \frac{x - x_m}{x_{m+1} - x_m}. \quad (\text{B.5})$$

### B.2 Lagrange Polynomial Interpolation (LPI)

LPI-4 is a fourth-order scheme and has four cubic polynomials as basis functions. For a point  $x$  lying in the interval  $[x_m, x_{m+1}]$  on a structured grid with constant grid spacing, the four basis functions are

$$\begin{aligned} b_1^x &= -\frac{1}{6}(\xi)(\xi - 1)(\xi - 2) \\ b_2^x &= \frac{1}{2}(\xi - 1)(\xi + 1)(\xi - 2) \\ b_3^x &= -\frac{1}{2}(\xi)(\xi + 1)(\xi - 2) \\ b_4^x &= \frac{1}{6}(\xi)(\xi + 1)(\xi - 1), \end{aligned} \quad (\text{B.6})$$

where  $\xi$  is the elemental coordinate defined as

$$\xi = \xi_1 = \xi_2 = \xi_3 = \xi_4 = \frac{x - x_m}{x_{m+1} - x_m}. \quad (\text{B.7})$$

These basis functions are nonzero over the entire interpolation stencil that spans the interval  $[x_{m-1}, x_{m+2}]$ . They are shown in Fig. 5.2(a).

### B.3 Piecewise Cubic Approximation (PCA)

PCA is a fourth-order scheme and has four *piecewise* cubic polynomials as basis functions. For a point  $x$  lying in the interval  $[x_m, x_{m+1}]$  on a structured grid with constant grid spacing,

the four basis functions are

$$\begin{aligned}
b_1^x &= \frac{1}{6}(2 + \xi_1)^3 & \xi_1 &= \frac{x_{m-1} - x}{h} \quad \text{for } -2 \leq \xi_1 < -1 \\
b_2^x &= \frac{1}{6}(-3\xi_2^3 - 6\xi_2^2 + 4) & \xi_2 &= \frac{x_m - x}{h} \quad \text{for } -1 \leq \xi_2 < 0 \\
b_3^x &= \frac{1}{6}(3\xi_3^3 - 6\xi_3^2 + 4) & \xi_3 &= \frac{x_{m+1} - x}{h} \quad \text{for } 0 \leq \xi_3 < 1 \\
b_4^x &= \frac{1}{6}(2 - \xi_4)^3 & \xi_4 &= \frac{x_{m+2} - x}{h} \quad \text{for } 1 \leq \xi_4 \leq 2,
\end{aligned} \tag{B.8}$$

where  $\xi_1, \xi_2, \xi_3, \xi_4$  are the elemental coordinates defined distinctively for each basis function. It is to be noted that unlike in LPI-4, the basis functions for PCA are defined only piecewise. Fig. 5.2(b) shows these basis functions which are nonzero in the interval  $[x_m, x_{m+1}]$ .

## B.4 Two Stage Estimation Algorithm

The TSE algorithm constructs a piecewise-polynomial approximation  $\tilde{\phi}(x)$  to a mean field  $\langle \phi(x) \rangle$  from particle data  $\phi^l$  given at locations  $X^l$ . It was originally developed by Dreeben and Pope (1992) for application to PDF methods, and has the advantage of working with unstructured grids also. It is being reproduced here for completeness. In this algorithm, the first stage constructs estimates at knots (center-of-mass locations of the particle data) using top-hat or linear basis functions (LPI-2). These first-stage estimates are then used as weighted data for the second stage in which a local least-squares algorithm is implemented to fit a linear or quadratic polynomial. The details for each stage are given in the next two subsections.

### B.4.0.1 Stage 1

The following quantities are defined in the first stage: The weight of the particles which support the  $m^{\text{th}}$  grid node,

$$w(x_m) = \sum_{c=m-1}^m \sum_{l=1}^{N^c} \mu^l b_{m-c+1}^x; \tag{B.9}$$

the center of mass of particles which support the  $m^{\text{th}}$  grid node,

$$\bar{X}(x_m) = \frac{\sum_{c=m-1}^m \sum_{l=1}^{N^c} X^l \mu^l b_{m-c+1}^x}{w(x_m)}; \tag{B.10}$$

and finally the estimate of the particle property at the center of mass,

$$\bar{\phi}(x_m) = \frac{\sum_{c=m-1}^m \sum_{l=1}^{N_c} \phi^l \mu^l b_{m-c+1}^x}{w(x_m)}, \quad (\text{B.11})$$

where  $\phi^l$  is called the particle property data. For mean interphase momentum transfer term estimation in  $x$ - direction, the expression for  $\phi^l$  is

$$\phi^l = f_x^l n(x_c, t),$$

where  $f_x$  is the particle force, and  $n(x_c, t)$  is the particle number density at the center of the  $c^{\text{th}}$  cell and it is computed as

$$n(x_c, t) = \frac{1}{V_c} \sum_{l=1}^{N_c} n^l. \quad (\text{B.12})$$

#### B.4.0.2 Stage 2

In stage 2, a local least-squares algorithm is implemented to calculate an approximation to the mean field that minimizes error with respect to the knot estimates. The output from the first stage,  $(\bar{X}(x_m), \bar{\phi}(x_m))$  along with the weights  $w_m$  forms the input for this stage. The objective of the local least-squares method is to provide an estimate for the mean field at the Eulerian grid node  $x_m$  by fitting a polynomial to data which lies within a neighborhood of  $x_m$ , the size of which is characterized by bandwidth,  $W$ . For each estimate, the data is weighted with a kernel  $Q$ , where

$$Q(u) \equiv \begin{cases} (1 - u^2)^2 & |u| \leq 1 \\ 0 & \text{otherwise.} \end{cases}$$

If  $x_p$  is an Eulerian grid node, then  $\tilde{\phi}(x_p)$  is a polynomial estimate for the underlying function in a neighborhood of  $x_m$  which minimizes the expression

$$\sum_m Q\left(\frac{\bar{X}(x_m) - x_p}{W}\right) w(x_m) [\tilde{\phi}(x_p) - \bar{\phi}(x_m)]. \quad (\text{B.13})$$

The linear two stage algorithm is implemented by fitting a first order polynomial to the points  $(\bar{X}(x_m), \bar{\phi}(x_m))$  in a neighborhood within a distance  $W$  centered at the grid node  $x_p$ . We take

a function of the form

$$\tilde{\phi}(x) = \tilde{a} + \tilde{b}(x - x_p),$$

where  $\tilde{a}$  and  $\tilde{b}$  are unknown constants to be determined. If

$$\hat{X}_{mp} = \bar{X}(x_m) - x_p \quad (\text{B.14})$$

$$\hat{Q}_{mp} = w(x_m) Q \left( \frac{\hat{X}_{mp}}{W} \right), \quad (\text{B.15})$$

then the constants  $\tilde{a}$  and  $\tilde{b}$  which minimize Eq. B.13 are determined by solving the matrix equation

$$\begin{bmatrix} \sum_m \hat{Q}_{mp} & \sum_m \hat{Q}_{mp} \hat{X}_{mp} \\ \sum_m \hat{Q}_{mp} \hat{X}_{mp} & \sum_m \hat{Q}_{mp} \hat{X}_{mp}^2 \end{bmatrix} \begin{bmatrix} \tilde{a} \\ \tilde{b} \end{bmatrix} = \begin{bmatrix} \sum_m \hat{Q}_{mp} \bar{\phi}_m \\ \sum_m \hat{Q}_{mp} \bar{\phi}_m \hat{X}_{mp} \end{bmatrix}. \quad (\text{B.16})$$

Finally, the estimate for the mean field - which is mean interphase momentum transfer term in our case - at the Eulerian grid node  $x_m$  is

$$\tilde{\phi}(x_m) = \tilde{a}. \quad (\text{B.17})$$

## APPENDIX C. DETAILS OF ERROR DECOMPOSITION

### C.1 Statistical Error

In the statistical error definition given by Eq. 5.22,  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  is an unbiased estimator of  $\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle$  but owing to finite sample size  $N_{pc}$ , it has statistical fluctuations measured by the variance  $\sigma_F^2$  of  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$ , which is given by Eq. C.1. The scaling of statistical error with number of samples is given by Eq. 5.23, where  $c_F$  which scales as  $\sigma_F$  is referred to as the statistical error coefficient, and  $\theta$  is a standardized normal variate.

$$\sigma_F^2 = N_{pc} \text{var}(\{\mathbf{F}_m^{fp}\}_{N_{pc},M}). \quad (\text{C.1})$$

In the definition of statistical error given by Eq. 5.22,  $\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle$  is an unknown and is approximated by ensemble averaging  $\{\mathbf{F}_m^{fp}\}_{N_{pc},M}$  over  $\mathcal{M}$  independent but identical simulations, such that

$$\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle \cong \langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle_{\mathcal{M}} = \frac{1}{\mathcal{M}} \sum_{i=1}^{\mathcal{M}} \{\mathbf{F}_m^{fp}\}_{N_{pc},M}^{(i)}. \quad (\text{C.2})$$

Note that  $\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle_{\mathcal{M}}$  is itself a random variable with mean and variance given by

$$\langle\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle_{\mathcal{M}}\rangle = \langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle. \quad (\text{C.3})$$

$$\text{var}(\langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle_{\mathcal{M}}) = \frac{1}{\mathcal{M}} \text{var}(\{\mathbf{F}_m^{fp}\}_{N_{pc},M}) = \frac{\sigma_F^2}{\mathcal{M}N_{pc}}. \quad (\text{C.4})$$

If the statistical error due to finite number of realizations  $\mathcal{M}$  is defined as

$$\Sigma_{F,\mathcal{M}} = \langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle_{\mathcal{M}} - \langle\{\mathbf{F}_m^{fp}\}_{N_{pc},M}\rangle, \quad (\text{C.5})$$

then from the central limit theorem, the scaling of  $\Sigma_{F,\mathcal{M}}$  with number of realizations is

$$\Sigma_{F,\mathcal{M}} = \mathcal{M}^{-1/2} \sigma_{F,\mathcal{M}} \xi, \quad (\text{C.6})$$

where  $\xi$  is a standardized normal variate, and  $\sigma_{F,\mathcal{M}}$  is the standard error which based on Eq. C.4 can be written as

$$\sigma_{F,\mathcal{M}}^2 = \mathcal{M}\text{var}(\langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}}) = \frac{\sigma_F^2}{N_{pc}} \quad (\text{C.7})$$

Therefore, the final form for  $\Sigma_{F,\mathcal{M}}$  is

$$\Sigma_{F,\mathcal{M}} = \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle = \frac{c_F \xi}{\sqrt{(\mathcal{M}N_{pc})}}. \quad (\text{C.8})$$

The total statistical error  $\Sigma_F$  can be rewritten as the summation of approximate statistical error denoted by  $\widehat{\Sigma}_F$  and  $\Sigma_{F,\mathcal{M}}$  (which is defined by above equation):

$$\Sigma_F = \widehat{\Sigma}_F + \Sigma_{F,\mathcal{M}}, \quad (\text{C.9})$$

where the definition of approximate statistical error and its scaling with the number of samples and realizations is given by

$$\begin{aligned} \widehat{\Sigma}_F &= \{\mathbf{F}_m^{fp}\}_{N_{pc},M} - \langle \{\mathbf{F}_m^{fp}\}_{N_{pc},M} \rangle_{\mathcal{M}} = \Sigma_F - \Sigma_{F,\mathcal{M}} \\ &= \frac{c_F \theta}{\sqrt{N_{pc}}} - \frac{c_F \xi}{\sqrt{(\mathcal{M}N_{pc})}}. \end{aligned} \quad (\text{C.10})$$

## APPENDIX D. DETAILS OF DDF EQUATION DERIVATION

### D.1 Evolution equation for $h$

In order to derive the evolution equation for  $h$ , we start by differentiating it with respect to time and noting that

$$\frac{\partial h}{\partial t} = \frac{\partial}{\partial t} \left\langle \sum_{i=1}^{N_c} h_i' \right\rangle = \left\langle \sum_{i=1}^{N_c} \frac{\partial h_i'}{\partial t} \right\rangle. \quad (\text{D.1})$$

Differentiating  $h_i'$  with respect to time results in

$$\begin{aligned} \frac{\partial h_i'}{\partial t} &= \frac{dW^{(i)}}{dt} \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} - \frac{\partial}{\partial x_k} \left\{ W^{(i)} V_{c,k}^{(i)} \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\} \\ &\quad - \frac{\partial}{\partial v_k} \left\{ W^{(i)} A_{c,k}^{(i)} \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\}. \end{aligned} \quad (\text{D.2})$$

Substituting for  $\frac{dW^{(i)}}{dt}$  (Eq. 6.17), the above equation becomes

$$\frac{\partial h_i'}{\partial t} = - \left\{ \Omega^{(i)} h_i' \right\} - \frac{\partial}{\partial x_k} \left\{ V_{c,k}^{(i)} h_i' \right\} - \frac{\partial}{\partial v_k} \left\{ A_{c,k}^{(i)} h_i' \right\}. \quad (\text{D.3})$$

Substituting the above equation in the evolution equation for  $h$  (Eq. D.1) results in

$$\frac{\partial h}{\partial t} = - \left\langle \sum_{i=1}^{N_c} \left\{ \Omega^{(i)} h_i' \right\} \right\rangle - \frac{\partial}{\partial x_k} [v_k h] - \frac{\partial}{\partial v_k} \left\{ \left\langle \sum_{i=1}^{N_c} A_{c,k}^{(i)} h_i' \right\rangle \right\}. \quad (\text{D.4})$$

We now define the following expressions in phase space

$$\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle = \frac{1}{h(\mathbf{x}, \mathbf{v}, t)} \left\langle \sum_{i=1}^{N_c} \left\{ \Omega^{(i)} h_i'(\mathbf{x}, \mathbf{v}, t) \right\} \right\rangle, \quad \text{if } h > 0 \quad (\text{D.5})$$

and equal to zero otherwise, and

$$\langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle = \frac{1}{h(\mathbf{x}, \mathbf{v}, t)} \left\{ \left\langle \sum_{i=1}^{N_c} A_{c,k}^{(i)} h_i'(\mathbf{x}, \mathbf{v}, t) \right\rangle \right\}, \quad \text{if } h > 0 \quad (\text{D.6})$$

and equal to zero otherwise. Substituting these definitions for conditional expectations in Eq. (D.4), we obtain the final evolution equation for  $h$  which is

$$\frac{\partial h}{\partial t} = - \langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle h - \frac{\partial}{\partial x_k} [v_k h] - \frac{\partial}{\partial v_k} \{ \langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle h \}. \quad (\text{D.7})$$

## D.2 Evolution equation for $\langle N_p(t) \rangle$

The rate of change of total number of physical particles is

$$\begin{aligned}
\frac{\partial \langle N_p(t) \rangle}{\partial t} &= \frac{\partial}{\partial t} \int_{[\mathbf{x}, \mathbf{v}]} h(\mathbf{x}, \mathbf{v}, t) d\mathbf{x} d\mathbf{v} = \int_{[\mathbf{x}, \mathbf{v}]} \frac{\partial h}{\partial t} d\mathbf{x} d\mathbf{v} \\
&= - \int_{[\mathbf{x}, \mathbf{v}]} \langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle h d\mathbf{x} d\mathbf{v} - \int_{[\mathbf{x}, \mathbf{v}]} \frac{\partial}{\partial x_k} [v_k h] d\mathbf{x} d\mathbf{v} \\
&\quad - \int_{[\mathbf{x}, \mathbf{v}]} \frac{\partial}{\partial v_k} \{ \langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle h \} d\mathbf{x} d\mathbf{v}.
\end{aligned} \tag{D.8}$$

The second and third terms in the above expression integrate to zero (Pope, 1985). On substituting the expression for conditional expectation of Omega  $\langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle$  given by Eq. (D.5) in the above equation results in

$$\begin{aligned}
\frac{\partial \langle N_p(t) \rangle}{\partial t} &= - \int_{[\mathbf{x}, \mathbf{v}]} \left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\rangle d\mathbf{x} d\mathbf{v} \\
&= - \left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \right\rangle = - \langle \Omega(t) \rangle \left\langle \sum_{i=1}^{N_c} W^{(i)}(t) \right\rangle \\
&= - \langle N_p(t) \rangle \langle \Omega(t) \rangle,
\end{aligned} \tag{D.9}$$

where the unconditional expectation of  $\Omega$  is defined as

$$\langle \Omega(t) \rangle = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \right\rangle}{\left\langle \sum_{i=1}^{N_c} W^{(i)}(t) \right\rangle} = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \right\rangle}{\langle N_p(t) \rangle}. \tag{D.10}$$

## D.3 Number density evolution equation

The evolution for number density  $\tilde{n}_p$  implied by  $h$  is obtained by integrating  $h$  over the  $\mathbf{v}$  space and then differentiating wrt  $t$  or

$$\begin{aligned}
\frac{\partial \tilde{n}_p(\mathbf{x}, t)}{\partial t} &= \frac{\partial}{\partial t} \int_{[\mathbf{v}]} h(\mathbf{x}, \mathbf{v}, t) d\mathbf{v} = \int_{[\mathbf{v}]} \frac{\partial h}{\partial t} d\mathbf{v} \\
&= - \int_{[\mathbf{v}]} \langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle h d\mathbf{v} - \int_{[\mathbf{v}]} \frac{\partial}{\partial x_k} [v_k h] d\mathbf{v} - \int_{[\mathbf{v}]} \frac{\partial}{\partial v_k} \{ \langle A_{c,k} | \mathbf{x}, \mathbf{v}; t \rangle h \} d\mathbf{v}.
\end{aligned} \tag{D.11}$$

The first term on the right hand side of the above equation becomes

$$\begin{aligned} \int_{[\mathbf{v}]} \langle \Omega | \mathbf{x}, \mathbf{v}; t \rangle h d\mathbf{v} &= \int_{[\mathbf{v}]} \left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\rangle d\mathbf{v} = \left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle \\ &= \langle \Omega | \mathbf{x}; t \rangle \left\langle \sum_{i=1}^{N_c} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle = \langle \Omega | \mathbf{x}; t \rangle \tilde{n}_p(\mathbf{x}, t), \end{aligned} \quad (\text{D.12})$$

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

$$\langle \Omega | \mathbf{x}; t \rangle = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle}{\left\langle \sum_{i=1}^{N_c} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle} = \frac{\left\langle \sum_{i=1}^{N_c} \Omega^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle}{\tilde{n}_p(\mathbf{x}, t)}, \quad \text{if } \tilde{n}_p > 0 \quad (\text{D.13})$$

and equal to zero otherwise.

The second term can be simplified as

$$\begin{aligned} \int_{[\mathbf{v}]} \frac{\partial}{\partial x_k} [v_k h] d\mathbf{v} &= \frac{\partial}{\partial x_k} \int_{[\mathbf{v}]} v_k \left\langle \sum_{i=1}^{N_c} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\rangle d\mathbf{v} \\ &= \nabla \cdot \left\{ \int_{[\mathbf{v}]} \left\langle \sum_{i=1}^{N_c} \mathbf{v}_c^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \delta_{\mathbf{v}_c^{(i)}} \right\rangle d\mathbf{v} \right\} \\ &= \nabla \cdot \left\{ \left\langle \sum_{i=1}^{N_c} \mathbf{v}_c^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle \right\} \\ &= \nabla \cdot \{ \langle \mathbf{V}_c(\mathbf{x}, t) \rangle \tilde{n}_p(\mathbf{x}, t) \}, \end{aligned} \quad (\text{D.14})$$

where the conditional expectation  $\langle \mathbf{V}_c(\mathbf{x}, t) \rangle$  is defined as

$$\langle \mathbf{V}_c(\mathbf{x}, t) \rangle = \frac{\left\langle \sum_{i=1}^{N_c} \mathbf{v}_c^{(i)} W^{(i)}(t) \delta_{\mathbf{x}_c^{(i)}} \right\rangle}{\tilde{n}_p}, \quad \text{if } \tilde{n}_p > 0 \quad (\text{D.15})$$

and equal to zero otherwise. Finally, the third term is zero (Pope, 1985). From the above equations, the final expression for the evolution of number density is

$$\frac{\partial \tilde{n}_p(\mathbf{x}, t)}{\partial t} + \nabla \cdot \{ \langle \mathbf{V}_c(\mathbf{x}, t) \rangle \tilde{n}_p(\mathbf{x}, t) \} = \langle \Omega | \mathbf{x}; t \rangle \tilde{n}_p(\mathbf{x}, t) \quad (\text{D.16})$$

#### D.4 Evolution of the velocity conditional pdf

The evolution equation for  $\tilde{h}_{\mathbf{V}_c}^c$  obtained by differentiating Eq. 6.18 with respect to time

$$\frac{\partial \tilde{h}_{\mathbf{V}_c}^c(\mathbf{v} | \mathbf{x}; t)}{\partial t} = \frac{1}{\tilde{n}_p(\mathbf{x}, t)} \frac{\partial h(\mathbf{v}, \mathbf{x}, t)}{\partial t} - \frac{h(\mathbf{x}, \mathbf{v}, t)}{\tilde{n}_p^2(\mathbf{x}, t)} \frac{\partial \tilde{n}_p(\mathbf{x}, t)}{\partial t}, \quad (\text{D.17})$$

followed by substituting the evolution equations for  $h$  (Eq. 6.21) and number density  $\tilde{n}_p$  (Eq. D.16), results in Eq. 6.27.

## Bibliography

- A. A. Amsden, and P. J. O'Rourke, and T. D. Butler (1989). KIVA-II: A Computer Program for Chemically Reactive Flows with Sprays. Technical Report LA-11560-MS, Los Alamos National Laboratory.
- Abraham, J. (1997). What Is Adequate Resolution in the Numerical Computations of Transient Jets? *SAE Paper* 970051.
- Allen, M. P. and Tildesley, D. J. (1989). *Computer Simulation of Liquids*. Oxford University Press, Oxford, United Kingdom.
- Amsden, A. A. (1993). KIVA-3: A KIVA Program with Block-Structured Mesh for Complex Geometries. Technical Report LA-12503-MS, Los Alamos National Laboratory, Los Alamos, NM 87545, USA.
- Anderson, T. B. and Jackson, R. (1967a). A fluid mechanical description of fluidized beds. *Ind. Eng. Chem. Fundam.*, 6:527-539.
- Anderson, T. B. and Jackson, R. (1967b). A fluid mechanical description of fluidized beds: Equations of motion. *I & EC Fundamentals*, 6(4):527-539.
- Aneja, R. and Abraham, J. (1998). How Far Does the Liquid Penetrate in a Diesel Engine: Computed Results Vs. Measurements. *Combustion Science and Technology*, 138(1-6):233-256.
- Are, S., Hou, S., and Schmidt, S. P. (2005). Second-order spatial accuracy in Lagrangian-Eulerian spray calculations. *Numerical Heat Transfer, Part B*, 48:25-44.

- Balachandar, S. and Maxey, M. R. (1989). Methods for evaluating fluid velocities in spectral simulations of turbulence . *J. Comp. Phys.*, **83**:96–125.
- Beetstra, R., van der Hoef, M. A., and Kuipers, J. A. M. (2007). Drag force of intermediate Reynolds number flows past mono- and bidisperse- arrays of spheres. *A.I.Ch.E.J.*, 53:489.
- Bhatnagar, P. L., Gross, E. P., and Krook, M. (1954). A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Physical Reviews*, 94:511–525.
- Bird, G. A. (1994). *Molecular gas dynamics and the direct simulation of gas flows*. Number 42 in Oxford engineering science series. Clarendon Press, Oxford.
- Boivin, M., Simonin, O., and Squires, K. D. (1998). Direct numerical simulation of turbulence modulation by particles in isotropic turbulence. *J. Fluid Mech.*, 375:235–263.
- Boivin, M., Simonin, O., and Squires, K. D. (2000). On the prediction of gas–solid flows with two–way coupling using large eddy simulations. *Phys. Fluids*, 12:2080–2090.
- Boyalakuntla, D. S. and Pannala, S. (2006). Summary of discrete element model (dem) implementation in mfix. Technical report, Oak Ridge National Laboratory, From URL <http://www.mfix.org/documents/MFIXDEM2006-4-1.pdf>.
- Boylakunta, D. J. (2003). *Simulation of Granular and Gas-Solid Flows using Discrete Element Method*. PhD dissertation, Carnegie Mellon University.
- Campbell, C. S. (1990). Rapid granular flows. *Annu. Rev. Fluid Mech.*, 22:57–92.
- Carte, T., Derksen, A., Portela, J. J., and Van Den Akker, H. E. A. (2004). Fully resolved simulations of colliding monodisperse sphere in forced isotropic turbulence. *J. Fluid Mech.*, 519:233–271.
- Chapman, S. and Cowling, T. G. (1953). *The mathematical theory of non-uniform gases*. Cambridge University Press, 2nd. edition.

- Chasnov, J. (1991). Simulations of the Kolmogorov inertial subrange using an improved subgrid model. *Phys. Fluids A*, 3:188–200.
- Chen, F., Drumm, E. C., and Guiochon, G. (2007). Prediction/Verification of Particle Motion in One Dimension with the Discrete-Element Method. *International journal of geomechanics*, 7:344–352.
- Chen, S. and Doolen, G. (1998). Lattice boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*.
- Clift, R., Grace, J. R., and Weber, M. E. (1978). *Bubbles, Drops and Particles*. Academic Press.
- Conte, S. D. and Boor, C. d. (1980). *Elementary Numerical Analysis: An Algorithmic Approach*, chapter 2, page 51. International Series in Pure and Applied Mathematics. McGraw Hill, 3rd edition.
- Crowe, C. T. (1982). Review—numerical models for dilute gas–particle flows. *ASME J. Fluids Engineering*, 104:297–303.
- Cundall, P. A. and Strack, O. D. L. (1978). The Distinct Element Method as a Tool for Research in Granular Media. Technical Report NSF Grant ENG76-20711, National Science Foundation.
- Cybulski, A., Van Dalen, M. J., Verkerk, J. W., and Van Den Berg, P. J. (1975). Gas-particle heat transfer coefficients in packed beds at low reynolds numbers. *Chemical Engineering Science*, 30:1015–1018.
- Denton, W. H. (1951). The heat transfer and flow resistance for fluid flow through randomly packed spheres, in general discussion on heat transfer. pages 370–373.
- Desjardins, O., Fox, R. O., and Villedieu, P. (2008). A quadrature-based moment method for dilute fluid-particle flows. *J. Comput. Phys.*, 227(4):2514–2539.

- Dreeben, T. D. and Pope, S. B. (1992). Nonparametric Estimation of Mean Fields with Application to Particle Methods for Turbulent Flows. Technical Report FDA 92-13, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853.
- Drew, D. A. (1971). Average field equations for two-phase media. *Stud. Appl. Math.*, 50:133–166.
- Drew, D. A. (1983). Mathematical modeling of two-phase flow. *Annu. Rev. Fluid Mech.*, 15:261–291.
- Drew, D. A. and Passman, S. L. (1998). *Theory of Multicomponent Fluids*. Applied Mathematical Sciences. Springer, New York.
- Elghobashi, S. E. and 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.
- Ergun, S. (1952). Fluid flow through packed columns. *Chem. Eng. Prog.*, 48:89–94.
- Evans, M. and Harlow, F. H. (1957). Hydrodynamic problems involving large fluid distortions. *J. Assoc. Comput. Mach.*, 4:137–142.
- Fan, R. and Fox, R. O. (2008). Segregation in polydisperse fluidized beds: Validation of a multi-fluid model. *Chemical Engineering Science*, 63(1):272 – 285.
- Fan, R., Marchisio, D. L., and Fox, R. O. (2004). Application of the direct quadrature method of moments to polydisperse gas-solid fluidized beds. *Powder Technology*, 139(1):7 – 20.
- Fox, R. O. (2003). *Computational Models for Turbulent Reacting Flows*. Cambridge University Press.
- Fox, R. O. (2008). A quadrature-based third-order moment method for dilute gas-particle flows. *Journal of Computational Physics*, 227:6313–6350.
- Frankl, F. I. (1953). On the theory of motion of suspended sediments. *Dokl. Akad. Nauk. SSSR*, pages 92–247.

- Garg, R., , Tenneti, S., and Subramaniam, S. (2009a). Scalar transport and heat transfer past ordered and random arrays of monodisperse spheres. part 2: finite reynolds number regime. *J. Fluid Mech.* Under review.
- Garg, R., Narayanan, C., Lakehal, D., and Subramaniam, S. (2007). Accurate numerical estimation of interphase momentum transfer in lagrangian-eulerian simulations of dispersed two-phase flows. *International Journal of Multiphase Flow*, 33:1337–1364.
- Garg, R., Narayanan, C., and Subramaniam, S. (2009b). A numerically convergent lagrangian-eulerian simulation method for dispersed two-phase flows. *International Journal of Multiphase Flow*, 35(4):376 – 388.
- Garg, R., Tenneti, S., Mohd-Yusof, J., and Subramaniam, S. (2009c). *Computational gas–solid flows and reacting systems: theory, methods and practice*, chapter Direct numerical simulation of gas–solid flow based on the immersed boundary method. IGI. In preparation.
- Garzó, V. and Dufty, J. W. (2002). Hydrodynamics for a granular binary mixture at low density. *Physics of Fluids*, 14(4):1476–1490.
- Garzo, V., Dufty, J. W., and Hrenya, C. M. (2007a). Enskog theory for polydisperse granular mixtures. i. navier-stokes order transport. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 76(3):031303.
- Garzo, V., Hrenya, C. M., and Dufty, J. W. (2007b). Enskog theory for polydisperse granular mixtures. ii. sonine polynomial approximation. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 76(3):031304.
- Glowinski, R., Pan, T. W., Hesla, T. I., Joseph, D. D., and Périaux, J. (2001). A Fictitious Domain Approach to the Direct Numerical Simulation of Incompressible Viscous Flow past Moving Rigid Bodies: Application to Particulate Flow. *Journal of Computational Physics*, 169(2):363–426.
- Goldstein, D., Handler, R., and Sirovich, L. (1993). Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics*, 105(2):354–366.

- Gunn, D. J. (1978). Transfer of heat and mass to particles in fixed and fluidized beds. *Intl. J. Heat Mass Transfer*, 21:467–476.
- Gunn, D. J. and Desouza, J. F. C. (1974). Heat-transfer and axial dispersion in packed-beds. *Chemical Engineering Science*, 29:1363–1371.
- Handley, D. and Heggs, P. J. (1968). Momentum and heat transfer mechanisms in regular shaped packings. *Transactions of the Institution of Chemical Engineers and the Chemical Engineer*, 46:T251–&.
- Harlow, F. H. (1988). PIC and its progeny. *Comp. Phys. Commun.*, 48:1–10.
- Hasimoto, H. (1959). on the periodic fundamental solutions of the Stokes equations and their application to viscous flow past a cubic array of spheres. *J. Fluid Mech.*, 5(2):317–328.
- Haworth, D. C. and Tahry, S. H. E. (1991). Probability density function approach for multi-dimensional turbulent flow calculations in reciprocating engines. *AIAA Journal*, 29(2):208–218.
- He, X. and Luo, L. (1997). Theory of the lattice boltzmann method: From the boltzmann equation to the lattice boltzmann equation. *Physical Review E*, 56:6811–6817.
- Hill, R. J., Koch, D. L., and Ladd, A. J. C. (2001a). The first effects of fluid inertia on flows in ordered and random arrays of spheres. *J. Fluid Mech.*, 448(213-241).
- Hill, R. J., Koch, D. L., and Ladd, A. J. C. (2001b). Moderate-Reynolds-number flows in ordered and random arrays of spheres. *J. Fluid Mech.*, 448(243-278).
- Hwang, W. and Eaton, J. K. (2006). Homogeneous and isotropic turbulence modulation by small heavy particles (St 50). *J. Fluid Mech.*, 564:361–393.
- Iddir, H., Arastoopour, H., and Hrenya, C. (2005). Analysis of binary and ternary granular mixtures behavior using the kinetic theory approach. *Powder Technology*, 151:117–125.

- Ikeno, T. and Kajishima, T. (2007). Finite-difference immersed boundary method consistent with wall conditions for incompressible turbulent flow simulations. *Journal of Computational Physics*, 226:1485–1508.
- Iyer, V. and Abraham, J. (1997). Penetration and dispersion of transient gas jets and sprays. *Combustion Science and Technology*, 130:315–334.
- Jaberi, F. A., Colucci, P. J., James, S., Givi, P., and Pope, S. B. (1999). Filtered mass density function for large-eddy simulation of turbulent reacting flows. *Journal of Fluid Mechanics*, 401:85–121.
- Jackson, R. (1983). *Theory of Dispersed Multiphase Flow*, chapter Some Mathematical and Physical Aspects of Continuum Models for the Motion of Granular Materials. Academic Press, New York.
- Jenkins, J. T. and Savage, S. B. (1983). A theory for the rapid flow of identical, smooth, nearly elastic, spherical particles. *J. Fluid Mech.*, 130:187–201.
- Kannenbergh, K. and Boyd, I. (2000). Strategies for efficient particle resolution in the direct simulation Monte Carlo method. *Journal of Computational Physics*, 157:727–745.
- Kashiwa, B. and Gaffney, E. (2003). Design Basis for CFDLib. Technical Report LA-UR-03-1295, Los Alamos National Lab.
- Kashiwa, B. and Rauenzahn, R. M. (1994). A Multimaterial Formalism. Technical Report LA-UR-94-771, Los Alamos National Lab.
- Kataoka, I. and Serizawa, A. (1989). Basic equations of turbulence in gas–liquid two–phase flow. *Intl. J. Multiphase Flow*, 15(5):843–855.
- Kim, J. and Moin, P. (1985). Application of a fractional-step method to incompressible Navier–Stokes equations. *Journal of Computational Physics*, 59:308–323.
- Koch, D. and Hill, R. (2001). Inertial effects in suspension and porous-media flows. Annual Review of Fluid Mechanics.

- Koch, D. L. (1990). Kinetic-theory for a monodisperse gas-solid suspension. *Physics of Fluids A-Fluid Dynamics*, 2:1711–1723.
- Kraichnan, R. (1976). Eddy viscosity in two and three dimensions. *Journal of Atmospheric science*, 33:1521–1536.
- Kunii, D. and Smith, J. M. (1961). heat transfer characteristics of porous rocks .2. thermal conductivities of unconsolidated particles with flowing fluids. *Aiche Journal*, 7:29–34.
- Ladd, A. and Verberg, R. (2001). Lattice-boltzmann simulations of particle-fluid suspensions. *Journal of Statistical Physics*.
- Ladd, A. J. C. (1994a). Numerical simulations of particulate suspensions via a discretized boltzmann-equation .1. theoretical foundation. *Journal of Fluid Mechanics*, 271:285–309.
- Ladd, A. J. C. (1994b). Numerical simulations of particulate suspensions via a discretized boltzmann-equation .2. numerical results. *Journal of Fluid Mechanics*, 271:311–339.
- Lakehal, D. and Narayanan, C. (2003). Numerical analysis of the continuum formulation for the initial evolution of mixing layers with particles. *Intl. J. Multiphase Flow*, 29:927–941.
- Lee, L. and Leveque, R. (2003). An immersed interface method for incompressible navier-stokes equations. *Siam Journal On Scientific Computing*, 25:832–856.
- Littman, H., Barile, R. G., and Pulsifer, A. H. (1968). Gas-particle heat transfer coefficients in packed beds at low Reynolds numbers. *Industrial & Engineering Chemistry Fundamentals*, 7:554–&.
- Moin, P. and Kim, J. (1982). Numerical investigation of turbulent channel flow. *J. Fluid Mech.*, 118:341–377.
- Monaghan, J. (1992). Smoothed particle hydrodynamics. *Annu. Rev. Astron. Astrophys.*, 30:543–574.
- Monaghan, J. and Lattanzio, J. (1985). A refined particle method for astrophysical problems. *Astron. Astrophys.*, 149:135–143.

- Moreau, M., Fede, P., and Simonin, O. V. P. (2003). Monte carlo simulation of colliding particles suspended in gas-solid homogeneous turbulent shear flows. volume 2 A, pages 491 – 500, Honolulu, HI, United States. Molecular chaos; Gas-solid flows.
- Narayanan, C., Lakehal, D., and Yadigaroglu, G. (2002). Linear stability analysis of particle-laden mixing layers using particle tracking. *Powder Technology*, 125:122–130.
- Okong'o, N. and Bellan, J. (2004). Consistent large-eddy simulation of a temporal mixing layer laden with evaporating drops. Part 1. Direct numerical simulation, formulation and *a priori* analysis. *J. Fluid Mech.*, 499:1–47.
- O'Rourke, P. and Amsden, A. A. (1987). The TAB Method for Numerical Calculation of Spray Droplet Breakup. *SAE Paper 872089*.
- Oseen, C. W. (1910). *Ark. Mat. Astron. Fys.*, 6:29.
- Pai, G. M. and Subramaniam, S. (2008). A comprehensive probability density function formalism for multiphase flows. *J. Fluid Mech.* Accepted, In Press.
- Pai, M. G. (2007). *Probability density function formalism for multiphase flows*. PhD dissertation, Iowa State University.
- Passalacqua, A., Garg, R., Subramaniam, S., and Fox, R. O. (2009). A fully coupled quadrature-based moment method for dilute fluid-particle flows. *Chemical Engineering Science*. In Review.
- Patankar, N. A. and Joseph, D. D. (2001). Lagrangian numerical simulation of particulate flows. *Intl. J. Multiphase Flow*, 27:1685–1706.
- Peskin, C. S. (1981). The fluid dynamics of heart valves: experimental, theoretical, and computational methods. *Annu. Rev. Fluid Mech.*, 14(235-259).
- Pfeffer, R. and Happel, J. (1964). Analytical study of heat and mass transfer in multiparticle systems at low reynolds numbers. *A.I.Ch.E. Journal*, 10(5):605 – 611.

- Piomelli, U. (1993). High Reynolds number calculations using the dynamic subgrid-scale stress model. *Phys. Fluids A*, 5:1484–1490.
- Plimpton, S. (1995). Fast parallel algorithms for short-range molecular-dynamics. *Journal of Computational Physics*, 117:1–19.
- Pope, S. B. (1985). PDF Methods for Turbulent Reactive Flows. *Prog. Energy Combust. Science*, 11:119–192.
- Pope, S. B. (1995). Particle method for turbulent flows: Integration of stochastic model equations. *J. Comp. Phys.*, 117(2):332–349.
- Pope, S. B. (2000). *Turbulent Flows*, chapter 1, pages 11–12. Cambridge University Press, Port Chester, NY.
- Raman, V., Pitsch, H., and Fox, R. O. (2005). Hybrid large-eddy simulation/Lagrangian filtered-density-function approach for simulating turbulent combustion. *Combustion and Flame*, 143:56–78.
- Ranz, W. E. and Marshall, W. R. (1952). Evaporation from drops. *Chem. Eng. Prog.*, 48:173–180.
- Savage, S. B. and Jeffrey, D. J. (1981). The stress tensor in a granular flow at high shear rates. *J. Fluid Mech.*, 110:255–272.
- Schiller, L. and Naumann, A. Z. (1933). A Drag Coefficient Correlation. *Z. Ver. Deutsch Ing.*, pages 318–320.
- Schmidt, D. P. and Rutland, C. J. (2000). A new droplet collision algorithm. *Journal of Computational Physics*, 164:62–80.
- Schumann, T. E. W. (1929). Heat transfer: A liquid flowing through a porous prism. *Journal of the Franklin Institute*, 208(3):405 – 416.

- Shen, J., Kaguei, S., and Wakao, N. (1981). Measurements of particle-to-gas heat-transfer coefficients from one-shot thermal responses in packed-beds. *Chemical Engineering Science*, 36:1283–1286.
- Shen, L., Wu, J., and Xiao, J. (2009). Experiments on chemical looping combustion of coal with a nio based oxygen carrier. *Combustion and Flame*, 156:721–728.
- Silbert, L., Ertas, D., Grest, G., Halsey, T., Levine, D., and Plimpton, S. (2001). Granular flow down an inclined plane: Bagnold scaling and rheology. *Physical Review E*, 64.
- Snider, D. M., O'Rourke, P. J., and 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.
- Snyder, L. J. and Stewart, W. E. (1966). velocity and pressure profiles for newtonian creeping flow in regular packed beds of spheres. *A.I.Ch.E.J.*, 12(1):167.
- Sorensen, J. P. and Stewart, W. E. (1974a). computation of forced-convection in slow flow through ducts and packed-beds .2. velocity profile in a simple cubic array of spheres. *Chemical Engineering Science*, 29(3):819–825.
- Sorensen, J. P. and Stewart, W. E. (1974b). Computation of forced-convection in slow flow through ducts and packed-beds .3. heat and mass-transfer in a simple cubic array of spheres. *Chemical Engineering Science*, 29:827–832.
- Squires, K. D. and Eaton, J. K. (1990). Particle response and turbulence modification in isotropic turbulence. *Phys. Fluids A*, 2:1191–1203.
- Squires, K. D. and Eaton, J. K. (1991). Preferential concentration of particles by turbulence. *Phys. Fluids A*, 3:1169–1178.
- Stoyan, D. and Stoyan, H. (1995). *Fractals, Random Shapes and Point Fields*. Wiley Series in Probability and Mathematical Statistics. John Wiley and Sons, New York.

- Subramaniam, S. (2000). Statistical representation of a spray as a point process. *Phys. Fluids*, 12(10):2413–2431.
- Subramaniam, S. (2001). Statistical modeling of sprays using the droplet distribution function. *Phys. Fluids*, 13(3):624–642.
- Subramaniam, S. and Haworth, D. C. (2000). A Pdf Method for Turbulent Mixing and Combustion on Three-Dimensional Unstructured Deforming Meshes. *Journal of Engine Research*, 1(2):171–190.
- Subramaniam, S. and O'Rourke, P. J. (1998). Numerical Convergence of the KIVA-3 Code for Sprays and its Implications for Modeling. Unclassified Report LAUR 98-5465, Los Alamos National Laboratory, Los Alamos, NM 87545.
- Subramaniam, S. and O'Rourke, P. J. (2001). Numerical Convergence of the KIVA-3 Code for Sprays and its Implications for Modeling. *Los Alamos National Laboratory: Los Alamos*.
- Sun, J., Battaglia, F., and Subramaniam, S. (2007). Hybrid two-fluid dem simulation of gas-solid fluidized beds. *Journal of Fluid Engineering*, 129(11):1394–1403.
- Sundaram, S. and Collins, L. R. (1996). Numerical considerations in simulating a turbulent suspension of finite-volume particles. *J. Comp. Phys.*, 124:337–350.
- Sundaram, S. and Collins, L. R. (1999). A numerical study of the modulation of isotropic turbulence by suspended particles. *J. Fluid Mech.*, 379:105–143.
- Syamlal, M. (1998). Mfix documentation: Numerical guide. Technical Report DOE/MC31346-5824, NTIS/DE98002029, National Energy Technology Laboratory, Department of Energy. See also URL <http://www.mfix.org>.
- Syamlal, M., Rogers, W., and O'Brien, T. J. (1993). Mfix documentation: Theory guide. Technical Report DOE/METC-95/1013, NTIS/DE95000031, National Energy Technology Laboratory, Department of Energy. See also URL <http://www.mfix.org>.

- Teletov, S. G. (1958). Problems of the hydrodynamics of two-phase mixtures. *I. Vestn. Mosk. Gos. Univ., Ser. Mat. Mekh. Astron. Fiz. Khim.*, 2:15–27.
- Turner, G. A. and Otten, L. (1973). Values of thermal (and other) parameters in packed-beds. *Industrial & Engineering Chemistry Process Design and Development*, 12:417–424.
- Tuzun, U., Houlsby, G. T., Nedderman, R. M., and Savage, S. B. (1982). The flow of granular materials—ii velocity distributions in slow flow. *Chemical Engineering Science*, 37(12):1691 – 1709.
- Uhlmann, M. (2005). An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209:448–476.
- Uijtewaal, W. S. J. and Oliemans, R. V. A. (1996). Particle dispersion and deposition in direct numerical and large eddy simulations of vertical pipe flows. *Phys. Fluids*, 8(10):2590–2604.
- van der Hoef, M. A., Beetstra, R., and Kuipers, J. A. M. (2005). Lattice-Boltzmann simulations of low-Reynolds-number flow past mono- and bidisperse arrays of sphere: results for the permeability and drag force. *J. Fluid Mech.*, 528:233–254.
- Verzicco, R., Mohd-Yusof, J., Orlandi, P., and Haworth, D. (2000). Large eddy simulation in complex geometric configurations using boundary body forces. *Aiaa Journal*, 38:427–433.
- Wakao, N. and Kaguei, S. (1982). *Heat and mass transfer in packed beds*, volume 1 of *Topics in chemical engineering*. Gordon and Breach science.
- Wakao, N., Kaguei, S., and Funazkri, T. (1978). Effect of fluid dispersion coefficients on particle-to-fluid heat transfer coefficients in packed beds. *Chemical Engineering Science*, 34:325–335.
- Wakao, N., Tanisho, S., and Shiozawa, B. (1977). thermal response of packed beds at low Reynolds numbers. *Heat Transfer - Japanese Research*, 6(4):56 – 60.
- Wang, Q. and Squires, K. (1996). Large eddy simulation of particle-laden turbulent channel flow. *Phys. Fluids*, 8:1207–1223.

- Weber, M. (2004). *Simulation of Cohesive Particle Flows in Granular and Gas-Solid Systems*. PhD dissertation, University of Colorado.
- Weber, M. W. and Hrenya, C. M. (2006). Square-well model for cohesion in fluidized beds. *Chemical Engineering Science*, 61(14):4511 – 4527. Mathematical modeling; Particles; Cohesion; Particle beds;.
- Welton, W. and Pope, S. B. (1997). PDF model calculations of compressible turbulent flows using smoothed particle hydrodynamics. *Journal of Computational Physics*, 134(1):150–168.
- Williams, F. A. (1958). Spray combustion and atomization. *Phys. Fluids*, 1(6):541–545.
- Wylie, J. J. and Koch, D. L. (2000). Particle clustering due to hydrodynamic interactions. *Physics of Fluids*, 12(5):964 – 970.
- Xu, J. and Pope, S. B. (1999). Assessment of numerical accuracy of PDF Monte Carlo methods for turbulent reacting flows. *J. Comp. Phys.*, 152(1):192–230.
- Xu, S. and Wang, Z. (2006). Systematic derivation of jump conditions for the immersed interface method in three-dimensional flow simulation. *Siam Journal On Scientific Computing*, 27:1948–1980.
- Xu, Y. (2008). *Modeling and Simulation of particle-laden turbulent flows*. PhD dissertation, Iowa State University.
- Yamamoto, K., Potthoff, M., Tanaka, T., Kajishima, T., and Tsuji, Y. (2001). Large-eddy simulation of turbulent gas-particle flow in a vertical channel: effect of considering inter-particle collisions. *J. Fluid Mech.*, 442:303–334.
- Yin, X. and Sundaresan, S. (2009). Drag law for bidisperse gas-solid suspensions containing equally sized spheres. *Industrial & Engineering Chemistry Research*, 48:227–241.
- Yueng, P. K. and Pope, S. B. (1988). An algorithm for tracking fluid particles in numerical simulation of homogeneous turbulence. *J. Comp. Phys.*, **79**:373–416.

- Yusof, J. M. (1996). *Interaction of Massive Particles with Turbulence*. PhD dissertation, Cornell University.
- Zick, A. A. and Homsy, G. M. (1982). Stokes flow through periodic arrays of spheres. *J. Fluid Mech.*, 115:13–26.
- Zinchenko, A. Z. (1994). Algorithm for random close packing of spheres with periodic boundary-conditions. *Journal of Computational Physics*, 114:298–307.

## ACKNOWLEDGEMENTS

It gives me great pleasure to acknowledge those individuals that have been instrumental in making my intellectual journey over the past six years most memorable. With sincerest gratitude, I would like to thank my major professor Prof. Shankar Subramaniam beginning from his decision to hire me as a graduate student to years of good advise, on both personal and professional fronts. I would like to thank Prof. Shankar Subramaniam for bearing with my odd working habits in my initial years and showing me the virtues of good work ethics and perseverance in research. I am deeply grateful to him for taking great efforts to enlighten me on the subtleties associated with the complex and challenging field of multiphase flows. I sincerely thank him for being an able mentor and also an honest friend.

I would like to express my heartfelt thanks to Profs. Rodney Fox for agreeing to serve on my committee and providing insightful suggestions on my research and dissertation. My warmest regards goes to Profs. Richard Pletcher, Michael Olsen, and Theodore Heindel for their invaluable time and their constructive comments on my work. I warmly thank Prof. Scott Hansen for agreeing to serve as my Applied Math Minor representative.

Many thanks go to my group mates Madhusudan Pai, Jun Sun, Ying Xu, Sergiy Markutsya, Vidyapati Tiwari, and Smita Kumari for their support and interesting discussions on research, course work, and recreational activities. I would like to specially thank Sudheer Tenneti with whom I worked very closely and successfully on the development of DNS code during the last two years of my PhD. During my stay at ISU, I have been fortunate to interact with my colleagues in the Computational Fluid Dynamics center, especially Anup Gokarn, Kolakaluri Ravi, Kiran Samudrala. I would like to thank Dr. Jim Coyle at high performance computing center for answering my endless queries on troubleshooting Linux operating system.

I'm deeply indebted to my parents who always choose education as the topmost priority in my growing up years, and for their inspiration and unrelenting support during these past years. My sincere gratitude goes to my brother Rohit Garg and my sister-in-law Aditi Agarwal for their support and encouragement.

This work was partially supported by the National Energy Technology Laboratory, US Department of Energy Grants No. DE-FC26-07NT43098 and DE-AC02-07CH11358.