ABSTRACT

SAINI, NADISH. Non-Spherical Particle Transport, Collision and Deposition in Turbulent Flows. (Under the direction of Dr. Clement Kleinstreuer.)

Particle-laden flows are pervasive in nature and industrial applications at both macroscopic and microscopic levels. Examples include dust and pollutant particles occurring in the environment, blood cells in arteries, therapeutic aerosols for direct inhalation, micronfibers used in cancer treatment, nanofluids for cooling and lubrication, carbon nanotubes for extra material strength, granular flow, pulverised coal-combustors, cyclone separators and pneumatic conveying of particles in industry, to name a few. Ellipsoids are the most widely studied non-spherical particles, not only because they have the analytically simplest non-spherical shape, but also because they are capable of representing a wide range of particle aspect ratios from spheres to elongated fibers which encompasses most smooth non-spherical particles.

Collision forces and torques, which are functions of the particle shape, are the most significant interactions governing the trajectory of particles and ultimately their deposition in two-phase flows. The discrete element method (DEM), which originated from molecular dynamics, is an accurate and efficient method for modelling the collision dynamics of particles. In the discrete element method the particles are modelled as distinct rigid bodies where the particle interior is not resolved. Typically for spherical particles collision forces and torques are calculated using the soft-sphere approach (Cundall and Strack, 1979), by representing particle overlap as a spring-mass-damper system. The soft-sphere approach is also adopted for non-spherical particles using the multi-sphere (MS) method. Introduced by Favier et al. (1999), the MS method is the current academic
and industrial standard for modelling the collision dynamics of non-spherical particles. It offers a mathematically simple and computationally rigorous algorithm. However, conflicting studies exist in the literature concerning the accuracy and efficiency of the MS approach to represent non-spherical particles (Lu et al., 2015).

With the main focus on turbulent particle-laden channel flows, the primary objective is to develop a novel algorithm for modelling the collision dynamics of non-spherical particles. A novel approach, i.e., ellipsoidal particle interaction (EPI) model, has been introduced and implemented in a 2-D turbulent particle-laden channel flow. A C++ based program has been developed to resolve the collision dynamics of EPI and to model the effects of hydrodynamics drag, lift and torque acting on non-spherical particles in turbulent flow. The results obtained are compared with existing computational data for 3-D ellipsoids. Agreement between the particle concentration model results from the EPI code and benchmark numerical data establishes the applicability of the EPI model for non-spherical particle collision. The proposed methodology is an improvement over the MS method in terms of accuracy and efficiency. Furthermore, the theoretical foundations have been established to extend the EPI model to 3-D ellipsoids with focus on mechanical and biomedical engineering applications.
© Copyright 2016 by Nadish Saini

All Rights Reserved
Non-Spherical Particle Transport, Collision and Deposition in Turbulent Flows

by
Nadish Saini

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Mechanical Engineering

Raleigh, North Carolina

2016

APPROVED BY:

______________________________  ______________________________
Dr. Andre Mazzoleni              Dr. Tarek Echekki

______________________________
Dr. Clement Kleinstreuer
Chair of Advisory Committee
DEDICATION

This work is dedicated to my parents, Mrs. Anu Saini and Mr. Sawroop Saini, for their unconditional love, support and courage. This work is a result of their resolve for providing me with the highest of education.
BIOGRAPHY

The author was born on June 10, 1990 in a small town, Dharamsala, in Himachal Pradesh, India. His high school education was from Sacred Heart High School, Sidhpur, India. He did his undergraduate studies in Mechanical Engineering from Punjab University, Chandigarh, India and graduated in 2012. He started his graduate studies at North Carolina State University, Raleigh, USA, in the Spring semester of 2014.
ACKNOWLEDGEMENTS

I would like to extend my deepest gratitude to my adviser, Dr. Clement Kleinstreuer, for his constant encouragement, guidance and support. He has been extremely supportive of my ideas and granted me the intellectual freedom which was necessary to achieve this work. He has been very patient and a source of great poise as I stumbled time and again through the course of implementation of this work.

I would also like to express my gratitude to Dr. Tarek Echekki and Dr. Andre Mazzoleni for taking time from their schedule and being a part of my thesis committee.

Deep appreciation is extended to Arun Varghese, Zelin Xu, Dr. Yu Feng and Mayank Vaish, members of the Computational Multi-Physics Lab, and Aditya Saini and Shreyas Narsipur, my fellow graduate students, for their constant guidance and healthy discussion sessions which have been instrumental in helping me achieving this work.

Special thanks to my parents, Mrs. Anu Saini and Mr. Sawroop Saini, for their unconditional love and support throughout the course of my education. Finally, I would also like to thank my good friends Dr. Sonalika Singh, Ishaan Sharma and Mohit Sharma for being a source of constant moral support and love in difficult times.
# TABLE OF CONTENTS

List of Tables ........................................................................................................... vii
List of Figures ............................................................................................................. viii

Chapter 1 Introduction and Literature Review ......................................................... 1
  1.1 Research Motivation ......................................................................................... 1
  1.2 Research Objectives ....................................................................................... 6
  1.3 Literature Review ........................................................................................... 7
    1.3.1 Computational Fluid Particle Dynamics (CF-PD) ................................. 8
    1.3.2 Multi-sphere (MS) Representation of Non-Spherical Particles .......... 13
    1.3.3 CFD-DEM Coupling Aspects: Hydrodynamic Forces and Torques .... 17
  1.4 Chapter Summary ........................................................................................... 31

Chapter 2 Dynamics of Non-Spherical Particles ..................................................... 33
  2.1 Lagrangian Phase Modelling ......................................................................... 35
    2.1.1 Coordinate Transformation .................................................................. 35
    2.1.2 Translational Dynamics ..................................................................... 38
    2.1.3 Rotational Dynamics ......................................................................... 42
  2.2 Numerical Integration of Particle Velocity and Position ................................. 44
  2.3 Collision Dynamics ....................................................................................... 49
    2.3.1 Contact Detection ............................................................................... 49
    2.3.2 Rough Wall and Rough Particle Modelling ....................................... 56
    2.3.3 Collision Forces and Torques .............................................................. 61
  2.4 Code Structure .............................................................................................. 71
  2.5 Chapter Summary ........................................................................................... 73

Chapter 3 Results and Discussion ......................................................................... 75
  3.1 Introduction .................................................................................................... 75
  3.2 Flow Field Solution ....................................................................................... 81
    3.2.1 Simulation Setup ............................................................................... 82
    3.2.2 Boundary conditions and Flow Initialization ..................................... 85
    3.2.3 Single-phase simulation results ......................................................... 86
  3.3 Particle-Laden Simulations .......................................................................... 89
    3.3.1 Mapping Eulerian Fields to Lagrangian Domain ............................... 89
    3.3.2 Simulation Setup ............................................................................... 92
    3.3.3 Smooth Wall Simulations: Spherical Particles .................................. 96
    3.3.4 Smooth Wall Simulations: Non-spherical Particles ......................... 102
    3.3.5 Rough-wall Collisions ..................................................................... 111
  3.4 Chapter Summary ........................................................................................... 117
LIST OF TABLES

Table 2.1 Values of coefficients used in the force and torque relations for different particle shapes (Zastawny et al., 2012) .......................... 45
Table 2.2 Relative position of two ellipses (Etayo et al., 2006) ................. 52
Table 3.1 Existing computational studies on particle-laden channel flow ... 77
Table 3.2 Different particles considered for analysis by van Wachem et al. (2015) 81
Table 3.3 Generated Mesh Specifications ........................................ 84
Table 3.4 Selected particle properties and time-step values for particle laden simulations ................................................................. 95
Table 3.5 Simulation run times (in clock secs) for different particles with EPI code ................................................................. 111
LIST OF FIGURES

Figure 1.1  Description of the hard-sphere methodology. Collision events are instantaneous ........................................... 11
Figure 1.2  Description of the soft-sphere methodology. Actual overlap between the particles is allowed ......................... 13
Figure 1.3  Contact between two particles composed of two sub-spheres each .......................................................... 15
Figure 1.4  Illustration of the transfer of forces acting on comprising sub-spheres to particle center ............................. 16

Figure 2.1  Definition of angle of incidence for non-spherical particles ................................................................. 41
Figure 2.2  Contact detection in multi-Sphere approach is reduced to finding overlap between comprising sub-spheres .................. 49
Figure 2.3  Illustration of Shadow effect due to wall roughness .............................................................................. 57
Figure 2.4  Modification of normal distribution function for different incident angles ................................................. 58
Figure 2.5  Virtual wall generation as proposed by Sommerfeld and Huber (1999) .................................................... 59
Figure 2.6  Plane along which the collision mechanics are modelled for smooth and rough spheres ............................... 60
Figure 2.7  (a) Spring mass damper system for modelling normal contact force. (b) Spring mass damper system for modelling tangential contact force. 61
Figure 2.8  Coefficient $\alpha$ as a function of restitution, as calculated by Tsuji et al. (1992) .................................................. 63
Figure 2.9  Collision point according to the (left) intersection and (right) geometric potential algorithm. The geometrical potential algorithm is also applicable to 3-D ellipsoids .............................................................................. 65
Figure 2.10 Collision point according to the common normal algorithm. Extendable to 3-D ellipsoids ................................. 65
Figure 2.11 The maxima and minima points on colliding ellipses after coordinate transformation ............................... 67
Figure 2.12 (a) Deformation in spheres during collision (b) Deformation in ellipses during collision and the selection of collision point. The dashed arcs represent the original shapes .................................................. 67
Figure 2.13 Calculation of deformation $\delta_n$ and the equivalent radius $r_p$ for calculation of collision forces. A similar process is repeated in the body coordinate system of particle $j$ to find $\delta_n$ and $r_p$ ............................................................................. 69
Figure 2.14 Sample space for the collision plane for rough ellipses and the corresponding location of center of interacting psuedo-sphere ........................................................................................................... 71
Figure 2.15 Flowchart of the EPI code .................................................................................................................. 74
Figure 3.1 Description of the test facility used by Kussin and Sommerfeld (2002) ................................................ 78
Figure 3.2 Geometry used in the simulations. Periodic boundary conditions (BC) are specified on the left-right and front-back faces.

Figure 3.3 Coarse mesh (Mesh 1 from Table 3.3) used for the simulations.

Figure 3.4 One-way simulation result comparison: Average stream-wise mean fluid velocity (in m/s).

Figure 3.5 Contours of instantaneous fluid velocity in the longitudinal cross-section of channel.

Figure 3.6 Depiction of bilinear interpolation of Eulerian quantities from cell centres to particle position D.

Figure 3.7 [Left] Near wall cells and the relative particle diameter. [Right] Combining the near wall cells to form a representative mesh. Mapping is based on cell area weighting.

Figure 3.8 Particle position with and without considering inter-particle collision at time $t = 0.5$ secs. (Particle sizes are magnified 5 times).

Figure 3.9 Results obtained from EPI code for spherical particles (considering smooth walls) for different number of particles introduced in domain.

Figure 3.10 Results obtained from EPI code for Ellipse 1, (Aspect Ratio = $\frac{5}{3}$, sphericity = 0.885) 400 particles (considering smooth walls).

Figure 3.11 Results obtained from EPI code for Ellipse 2, (Aspect Ratio = $\frac{5}{4}$, sphericity = 0.991) 400 particles (considering smooth walls).

Figure 3.12 Results obtained from EPI code for Fiber, (Aspect Ratio = 5, sphericity = 0.639) 400 particles (considering smooth walls).

Figure 3.13 Comparison of the results obtained by EPI code for different particle shapes.

Figure 3.14 Comparison of the average angular velocity of the particles obtained from the EPI code.

Figure 3.15 Average angular velocity trend reported by Zhao and van Wachem (2013a) using DNS one-way simulation of 3-D ellipsoid particles with aspect ratio = 3. The profile obtained from the EPI code follows a similar trend.

Figure 3.16 Results obtained from EPI code for Ellipse 1, (Aspect Ratio = $\frac{5}{3}$, sphericity = 0.885) 400 particles (considering rough walls, $\gamma = 18^\circ$).

Figure 3.17 Results obtained from EPI code for Ellipse 2, (Aspect Ratio = $\frac{5}{4}$, sphericity = 0.991) 400 particles (considering rough walls, $\gamma = 18^\circ$).

Figure 3.18 Results obtained from EPI code for Fiber, (Aspect Ratio = 5, sphericity = 0.639) 400 particles (considering rough walls, $\gamma = 18^\circ$).

Figure 3.19 Comparison of the results obtained by EPI code for different particle shapes (considering rough walls, $\gamma = 18^\circ$).

Figure 4.1 Proposed flow-chart for the development of 3-D fully coupled EPI model.
Figure 4.2 Description of the algorithm for calculating the interacting pseudo-sphere for 3-D ellipsoid collision .................. 130
Chapter 1

Introduction and Literature Review

1.1 Research Motivation

Particle laden flows are pervasive in nature and industrial applications at both macroscopic and microscopic levels. Dust and pollutant particles occurring in the environment, blood cells in arteries, therapeutic aerosols for direct inhalation, micron-fibers used in cancer treatment, nanofluids for cooling and lubrication, carbon nanotubes for extra material strength, granular flow, pulverised coal-combustors, cyclone separators and pneumatic conveying of particles in industry, to name a few examples. Therefore it is imperative to study the mechanics of transport of particles in suspended in fluid flow. Clearly, one of the major factors that influences the trajectory of particles is the particle shape.

It is intuitively obvious that non-spherical particles will have significantly different pre-collision and post-collision trajectories when compared to spherical particles. Furthermore, the hydrodynamic drag and lift forces acting on non-spherical particles are also vastly different, leading to different trajectories in a given flow domain, even when
inter-particle collisions are not accounted for. Nevertheless, several studies have shown that the dynamics of real non-spherical particles, such as granular flow (Rothenburg and Bathurst, 1991; Campbell, 2011) or toxic air pollutants and therapeutic drug aerosols (Feng and Kleinstreuer, 2013), can be estimated via sphere-equivalent correlations or by considering them to be ellipsoids. An alternative approach, but very taxing on computer resources, is direct numerical simulation where the pressure and stress fields around the particles are directly calculated at all times (Shenoy and Kleinstreuer, 2008, 2010). Ellipsoids are the most widely studied non-spherical particles, not only because they have the analytically simplest non-spherical shape, but also because they are capable of representing a wide range of particle aspect ratios from spheres to elongated fibers which encompasses most smooth non-spherical particles.

Most of the existing experimental (Lee and Durst, 1982; Rogers and Eaton, 1990; Kulick et al., 1994; Kussin and Sommerfeld, 2002; Lam et al., 2002; Eaton, 2009) and computational research (Wang and Squires, 1996; Yamamoto et al., 2001; Kuerten, 2006; Marchioli et al., 2008; Mallouppas and van Wachem, 2013) on particulate flows has been done considering the particles to be spherical. Note that even when considering only spheres a large number of phenomena need to be taken into account for correct trajectory calculations (Sommerfeld, 2003). On the other hand no experimental studies on non-spherical particles were found in the open literature. Computational studies on non-spherical particles have begun to emerge very recently and have been focused mostly on ellipses/ellipsoids. Naturally, data on spherical particles is used as a benchmark for non-spherical computations (e.g., van Wachem et al. (2015)). Considerable research has been done to define relations for the drag force acting on non-spherical particles (Jeffery, 1922; Brenner, 1964; Rosendahl, 2000; Hölzer and Sommer-
feld, 2008, 2009; Loth, 2008). However for non-spherical particles the lift force (Hoerner, 1965; Mandø and Rosendahl, 2010; Harper and Chang, 1968) and hydrodynamic torques (Jeffery, 1922) acting on the particle also become significant. Fewer studies have explored these effects on non-spherical particles and they too rely heavily on empirical data, frequently with relations that are valid only for low Reynolds number flows. In contrast, Zastawny et al. (2012) performed DNS simulation on some particular aspect ratios of ellipsoids to derive drag and lift force as well as pitching and resisting torques. Although the applicability of these relations for all particle aspect ratios was not discussed, the study does provide a general framework for constructing these relations. In fact, they are used in the present study when the particle aspect ratios being considered are the same for which the hydrodynamic relations were derived.

There are two major methods for modelling collision mechanics of non-spherical particles i.e., the single-particle approach and the composite sphere approach. A thorough discussion of the two approaches has been presented by Lu et al. (2015). Introduced by Favier et al. (1999), the composite-sphere approach, known as the multi-sphere (MS) approach is the current academic and industrial standard for the discrete element method (DEM). It has been implemented in commercial packages such as PFC (2016) and EDEM (2016). In the MS approach a non-spherical particle is represented by an agglomeration of several spherical particles of the same or different diameters. The spherical particles are ‘glued’ together such that their relative position remains unchanged throughout the simulation. The forces and torques acting on comprising spheres during a collision event are accumulated relative to the center of mass of the composite particle via vectorial summation and then used to calculate the trajectory of the composite particle. For spherical particles, contact detection as well as contact point and plane calculation algorithms are
very simple. Because in the MS approach collision of non-spherical particles is reduced
to collision of spheres, the implementation of the MS approach is rather straightforward
and easy to program; hence it’s popularity. However, in order to accurately represent
the shape of a non-spherical particle a very large number of comprising spheres may
be required. Conflicting studies exist in the literature which focus on the accuracy of
the MS approach to represent non-spherical particles (Lu et al., 2015). For example,
Kruggel-Emden et al. (2008) conducted simulations on particle-wall interactions using
the MS approach, where a spherical particle was composed of different sets of comprising
spheres i.e., 7 to 300 spheres. They compared the obtained numerical data with simu-
lations where considering the particle to be a single sphere and with the experimental
investigation conducted by Gorham and Kharaz (2000). They concluded that while the
simulation with a single sphere matched very well with the experimental outcome, the
results for the MS approach were highly unreliable due to large errors, depending on the
initial release orientation of the MS particle. They also concluded that there was no direct
correlation between the number of comprising spheres used to model an MS particle and
the measured numerical accuracy. Since the MS approach had proven limitations when
used for approximating a spherical particle, they conjectured that it is also expected to
depict non-spherical particles erroneously.

A more recent study by Kačianauskas et al. (2014) considered the impact characteris-
tics of randomly shaped particles. The surface profile of the particle was described using
a probability distribution function. They reported that the dependency of the contact
characteristics of the particle had no correlation to the increasing number of comprising
spheres used to model a particle, not only in a deterministic sense but also statistically.
In contrast to the above discussed studies, the question of applicability of the MS approach was addressed by Markauskas et al. (2010) who conducted simulation experiments whose results supported the modelling technique. This article is still being used as a key reference for the veracity of the MS approach by a large number of studies dealing with non-spherical particles. Specifically, Markauskas et al. (2010) demonstrated the capability of the MS approach to model an assembly of ellipsoids. The approach was applied to the "piling problem" of 1000 vertically released ellipsoids. The ellipsoids were approximated using 3, 5, 7, 9, 13 and 17 spheres. They analyzed both the macroscopic, i.e., repose angle of the pile and the microscopic properties i.e., porosity of the pile and the coordination number. For comparison, 2-D ellipses were simulated where the contact forces were resolved using the exact analytical equation of ellipses. They found that the results for the two simulations matched closely; thus, establishing the applicability of MS method to non-spherical DEM.

A possible reason for the differences in the conclusions made by Kruggel-Emden et al. (2008) (or Kačianauskas et al. (2014)) and Markauskas et al. (2010) is that the latter analyzed the granular type or bulk-flow of particles, whereas the former ones studied collision characteristics of individual particles. In bulk-flow type scenarios the errors arising in a single collision are compensated for by simultaneous multiple contacts of the particles such that the trajectory of the particle is characterized only by the overall dimensions of the particle (major and minor axis of the ellipsoid). Thus, it can be argued that while MS approach is sufficiently accurate in simulating granular type flows where particles are in constant contact, it may not be accurate for simulation of particle-laden flows where the particles are not 'sticking' to each other. Further investigations are being conducted to prove this conjecture.
This project focuses on the simulation of elliptical particles in turbulent channel flows. Section 1.2 lists the novel research objectives based on the literature review. Chapter 2 outlines the methodology used and also introduces a new collision modelling technique based on pseudo-spheres for elliptical particles. The approach is termed Ellipsoidal Particle Interaction (EPI) modelling. Chapter 3 provides the results obtained for turbulent channel flows using the EPI model. Finally conclusions drawn from the results and the future developments of the EPI technique as well as future work are discussed in Chapter 4.

1.2 Research Objectives

Based on the discussions presented in Section 1.1 development of a more accurate discrete element (DEM) method for modelling non-spherical particle collision is currently a requirement for various applications, especially in mechanical and biomedical engineering. Thus, the research objectives are:

1. A thorough literature review of the existing DEMs for spherical and non-spherical particles as well as the existing experimental and computational studies on particle-laden flows.

2. Development of a (2-D) collision theory based on the DEM techniques for ellipsoids of arbitrary aspect ratios as well as the numerical implementation of the collision theory, documented in a computer program.

3. Integration of the collision model into the CFD code, i.e., development of a one-way coupled CFD-DEM (2-D) code to model the trajectory of particles in a (turbulent or laminar) flow field.
4. Verification of the developed one-way particle trajectory code via steady particle-laden channel flow simulations, matching the work of Mallouppas and van Wachem (2013) and van Wachem et al. (2015).

5. Thorough investigation of the ellipsoidal particle behavior in turbulent channel flows.

6. Preparations towards an extension to 3-D ellipsoidal particle collision, transport and deposition with mechanical and biomedical engineering applications.

1.3 Literature Review

The mechanics of real matter can be classified into two main branches i.e., Lagrangian and Eulerian. In the Lagrangian framework an individual parcel of matter is followed as it moves through space and time. In contrast, Eulerian specification focuses on specific locations in space and observes the flux of matter, momentum and energy through that space in time. Fluid flow is typically treated with the Eulerian specification, because it is prohibitive to follow each individual molecule of fluid. Lagrangian mechanics is used where the individual particles are large in size such that it is physically and computationally possible to track the particles. Since particles considered are discrete, inter-particle collisions also need to be accounted for in Lagrangian dynamics. Several engineering applications involve simultaneous flux of fluid and large rigid particles, such as in particle suspension flows. Therefore, it is important to comprehend how the Lagrangian and Eulerian mechanics interact with each other and how can they be modelled simultaneously. The following sections introduce the details of both the modelling mechanisms and also discuss how coupling between the two is achieved. Particular methods to model
interactions between spherical and non-spherical Lagrangian particles are also detailed.

### 1.3.1 Computational Fluid Particle Dynamics (CF-PD)

#### 1.3.1.1 Large Eddy Simulation (LES)

In Eulerian or continuum mechanics the equations governing the conservation of mass, momentum and energy are the Navier-Stokes equations. The equations in tensor notation are

\[
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_j} = 0 \tag{1.1}
\]

\[
\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \mu \left( 2S_{ij} - \frac{2}{3} \delta_{ij} S_{kk} \right) \tag{1.2}
\]

\[
\frac{\partial e}{\partial t} + \frac{\partial (u_j e)}{\partial x_j} = \frac{\partial Q}{\partial t} - \frac{\partial}{\partial x_j} \left( \kappa \frac{\partial T}{\partial x_j} \right) \tag{1.3}
\]

where \( u_i \) is the velocity, \( p \) is the pressure, \( \rho \) is the density of the fluid, \( \mu \) is the dynamic viscosity of the fluid, \( T \) is the temperature, \( \frac{\partial Q}{\partial t} \) is the heat generation per unit volume, \( \kappa = \frac{\alpha}{\rho c_p} \) is the molecular conductivity, \( \alpha \) is the thermal diffusivity. \( e \) is the total energy per unit volume,

\[
e = c_v T + \rho u_i u_i / 2 \tag{1.4}
\]

c\( _p \) is the specific heat at constant pressure and \( S_{ij} \) is the strain tensor,

\[
S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{1.5}
\]

In a turbulent flow the length of the scales may vary from the Kolmogorov scale to
very large fluid structures. The Kolmogorov scales are the smallest length and time scales in turbulence are defined, respectively, as,

\[ \eta = \left( \frac{\nu^3}{\epsilon} \right)^{\frac{1}{4}} \]  
\[ \tau_\eta = \left( \frac{U'}{\epsilon} \right)^{\frac{1}{2}} \]  

where \( \nu \) is the kinematic viscosity of the fluid and \( \epsilon \) is the average rate of dissipation of turbulence kinetic energy per unit mass. The Navier-Stokes equations incorporate all length and time scales but it is computationally prohibitive and often not required to resolve all scales of turbulence depending on the application. Thus a filter is applied to the Navier-Stokes equations which separates the large scales from the smaller ones. It is defined as Piomelli (1997)

\[ \tilde{f}(x) = \int_D f(x')G(x,x')dx' \]  

where \( D \) denotes the domain; \( G \) is the filter function and \( x' \) is the fluctuating component of any variable \( x \). Some of the commonly used filters are the sharp Fourier cutoff filter, Gaussian filter and the tophat filter (see Piomelli (1997)). The purpose of the filters is basically to define the cut-off length below which the turbulent scales are not resolved. In LES simulation the scales smaller than the cut-off length are modelled by the so-called subgrid scale models (SGS). The filtered incompressible Navier-Stokes equations are written as

\[ \frac{\partial \bar{u}_i}{\partial x_i} = 0 \]  
\[ \frac{\partial \bar{u}_i}{\partial t} + \frac{\partial (\bar{u}_i \bar{u}_j)}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_i \partial x_j} \]
1.3.1.2 Discrete Element Method (DEM) technology

The fundamental methodology of the discrete element method (DEM) is that the particles are modelled as distinct rigid bodies such that the particle interior is not resolved. The shape of the particle is considered invariant and their interaction is modelled by either treating the particles as completely rigid (hard sphere method) or as ‘soft’ particles with elastic properties (soft sphere method). The DEM approach avoids the computational effort of discretizing the actual particle and modelling its mechanics by methods such as finite element method (FEM). The DEM technology originated from molecular dynamics (MD) simulations and has been successfully adapted to macroscopic particles. In MD the particles are mostly treated as hard spheres, however for macroscopic particles the hard sphere methodology is not very accurate.

1.3.1.2.1 Comparison of Hard and soft sphere approaches  The hard sphere approach was first introduced by Maw et al. (1976). In this method the particles are considered to be perfectly rigid and the contact between two colliding particles is regarded as instantaneous. Therefore the final velocities after collision are modelled using the conservation principles of linear and angular momentum (see Figure 1.1). The final velocities rely on collision parameters such as coefficient of restitution, coefficient of friction, coefficient of moment restitution. Realistically these parameters often change during the course of collision and are also functions of input velocities. They have to determined empirically and then input to the hard sphere method for collision modelling. For non-spherical particles the equations resulting from the hard sphere approach are too large and unwieldy. Brach (1989) presented detailed derivations for the collision equations of arbitrarily shaped 2-D particles. The equations for 3-D collision, however, were not dis-
cussed owing to their formidable complexity. Thus the hard sphere approach can not be adopted directly for ellipsoids or any non-spherical particle in general.

Figure 1.1: Description of the hard-sphere methodology. Collision events are instantaneous.

The soft sphere method was first successfully applied to granular flow by Cundall and Strack (1979). The normal collision forces were modelled as a function of the overlap, $\delta_{ij}$ between two spheres (see Figure 1.2). The mathematical details of the soft-sphere approach are presented in Chapter-2, Section 2.3.3. Tsuji et al. (1992) extended the analysis to model tangential forces and torques acting during collision in a similar way. All the forces and torques depend on the mechanical properties of the particles i.e., Young’s modulus and Bulk modulus. This offers a major advantage over the hard sphere approach.
since the empirical factors are avoided and is a major reason that this approach became the standard method for DEM simulations.

Further, the different mathematical approaches of hard and soft sphere approach dictates very different computer algorithms for the two. Hard sphere method is implemented with an event drive (ED) algorithm (Donev et al., 2005). Since the collision is instantaneous, the collision event between two particles is predicted beforehand. The simulation time is advanced to that particular instant and the collision mechanics implemented before advancing to next collision event. Note that owing to the mechanism of ED algorithm hard sphere model is capable of modelling only binary particle collisions. On the other hand the soft sphere method uses a time driven (TD) algorithm (Donev et al., 2005). The simulation time moves at fixed time step such that the entire collision event is resolved in time. Since realistic collision duration in general are very small, the TD algorithm requires that the simulation time step is very small. Thus, simulations using the soft sphere approach are more computationally intensive as compared to hard sphere approach.

Mallouppas and van Wachem (2013) applied both hard and soft sphere approaches to particle laden turbulent flow simulations and compared the results obtained from both with experimental results reported by Kussin and Sommerfeld (2002). They pointed out that the suitability of hard sphere approach is limited to fast moving dilute flow suspensions. For dense flows the soft sphere approach is more accurate since it is based only on the mechanical properties of the particles. For the concentration considered by Mallouppas and van Wachem (2013) both the approaches were applicable. They reported that the particle based results matched with the experimental data better using the soft sphere approach and attributed this effect to the dependence of hard sphere approach on
empirical properties.

Figure 1.2: Description of the soft-sphere methodology. Actual overlap between the particles is allowed

1.3.2 Multi-sphere (MS) Representation of Non-Spherical Particles

Research on non-spherical particle collision mechanics had not picked up until recently due to the complexities involved in representing the surface of particles by analytical relations. Favier et al. (1999), Jensen et al. (1999) and Vu-Quoc et al. (2000) introduced the so-called Multi-Sphere approach to model non-spherical particles of any shape. They all introduced the MS approach as applied to ellipsoidal particles, through different mecha-
nisms. Since then the MS approach has become an industrial and academic standard for simulating non-spherical particles.

Favier et al. (1999) provided the fundamentals for mapping the forces acting on comprising spheres onto the parent non-spherical particle. For simplicity, consider two representative ellipsoid particles undergoing collision as shown in Figure 1.3. Each ellipsoid comprises of only two sub-spheres. Calculation of the contact point $c$ is straightforward for two overlapping spheres. The relative position vector from center of ellipsoid to the sub-sphere center, $d_{ps}$ remains constant. $r_{p(g)}$ is the global position vector of ellipsoid center and $r_{ps}$ is the global position vector of sub-sphere center. $r_{psc}$ is the relative vector from sub-sphere center to collision point and $l_{pc}$ is the relative vector from ellipsoid center to collision point. Collision force are calculated for the overlapping sub-spheres by the relations put forth by Tsuji et al. (1992). These forces are applied at the the contact point $c$,

$$F_c = f_t + f_n$$  \hspace{1cm} (1.11)

where $F_c$ is the total force acting at the contact point and $F_t$ and $F_n$ are the calculated tangential and normal forces respectively. Figure 1.4a shows the forces acting on three different contact points at a particular instant, marked as $pA_1$, $pA_2$ and $pB_1$. The resultant contact force acting at the center of each sub-sphere is the vector sum of forces acting on contact points. Thus,

$$f_{ps} = \sum_{c=1}^{c} f_{psc}$$  \hspace{1cm} (1.12)
where $s$ is used as particle index and $c$ is used as contact point index. Figure 1.4b shows a visual representation of the resulting forces. Finally the total out-of-balance force acting on the ellipsoid is given by the vectorial sum of resultant forces acting on its comprising sub-spheres.

$$f_p = \sum_{s=1}^{s} f_{ps}$$  \hspace{1cm} (1.13)

Figure 1.3: Contact between two particles composed of two sub-spheres each
Figure 1.4: Illustration of the transfer of forces acting on comprising sub-spheres to particle center.

The total moment about the center of a comprising sub-sphere due to tangential force
is given by

\[ M_{tp} = \sum_{c=1}^{c} (r_{psc} \times f_{psc}) \]  (1.14)

where \( f_{psc} \) is the tangential component of the contact force at point \( c \). Forces acting at the center of the comprising sub-spheres that do not pass through the center of ellipsoid add another moment to the ellipsoid. Finally the total moment acting on the ellipsoid can be written as,

\[ M_p = \sum_{s=1}^{s} [d_{ps} \times f_{ps} + M_{tp}] \]  (1.15)

As evident MS method offers an easy to implement algorithm for the dynamics of non-spherical particles, for contact point calculation, contact detection and calculation of collision forces and torques.

1.3.3 CFD-DEM Coupling Aspects: Hydrodynamic Forces and Torques

For particle-laden fluid flows three important aspects need to be addressed for a successful simulation,

1. Describe models for the calculation of particle-particle and particle wall interactions as they play an important role in particle trajectory and turbulence modification of the fluid flow.

2. Describe theories for converting Eulerian fields to Lagrangian and vice-versa. The particle are treated as discrete point masses and their motion is governed by New-
ton’s laws of motion, described in the Lagrangian framework, whereas the fluid motion is governed by continuum mechanics and its motion is governed by the Navier-Stokes equation in the Eulerian framework.

3. Coupling of the discrete and continuum frameworks.

Collision model for particles are introduced in previous sections and discussed in detail in Chapter 2. The Lagrangian parameters are accounted for in the Navier-Stokes equations by means of a source term addition to the equation. Thus equation (1.10) is modified as,

\[
\frac{\partial \alpha_f u_f}{\partial t} + \frac{\partial \alpha_f u_f v_f}{\partial x_i} = -\frac{1}{\rho} \frac{\partial \rho}{\partial x_j} + \frac{\partial (\alpha_f \tau_{ij})}{\partial x_i} + \beta_{(f,p)} (\bar{v}_{f\rho} - \bar{v}_{p}) \tag{1.16}
\]

where the last term on the right hand side of above equation is the momentum exchange term which provides coupling from Lagrangian field. \(\alpha_f\) is the fluid volume fraction in the control volume, \(\bar{v}_{f\rho}\) is undisturbed the velocity of the fluid at the position of the particle, \(\bar{v}_{p}\) is the velocity of the particle in the Eulerian framework.

On the other hand, the effect of the fluid flow on the particle is accounted for by Newton’s second law,

\[
m_p \frac{d\vec{v}_p}{dt} = \vec{F}_D + \vec{F}_L + \vec{F}_{VM} + \vec{F}_h + \vec{F}_g + \vec{F}_{pp} + \vec{F}_{pw} + \vec{F}_b \tag{1.17}
\]

where \(\vec{F}_D\) is the drag force; \(\vec{F}_P\) is the force due to the pressure gradient; \(\vec{F}_{VM}\) is the virtual mass force; \(\vec{F}_g\) is the net gravitational force; \(\vec{F}_L\) is the lift force; \(\vec{F}_{pp}\) is the force due to interaction of particles; \(\vec{F}_{pw}\) is the force due to interaction of a particle with boundary; \(\vec{F}_h\) is the Basset force due to history effects and \(\vec{F}_b\) is the force due to buoyancy effects. Out of all the forces listed that may act on the particle, drag and lift force are the ones that
act on the particle due to fluid interaction and serve for coupling of Eulerian fields to Lagrangian. Fluid particle flows may be classified as dense or dilute based on the forces that dominate the trajectory of particles. In dilute flows body forces (gravity, virtual mass) and surface forces (pressure, drag, lift) dominate whereas in dense flows inter-particle collision forces are more significant due to frequent interactions. Full coupling between the Eulerian and Lagrangian phases may not be required for all flows. A one-way coupled flow is where only the effect of the fluid on the particles is accounted for. In a two-way coupled flow the effect of the particle on the fluid is also included via the momentum exchange term. In a fully coupled, or also known as four-way coupled, flow the inter-particle interactions are also included.

An important scaling parameter in particulate flows is the Stokes number define as,

\[ St = \frac{\tau_R}{\tau_F} = \frac{\rho_p d_p^2 U}{18 \mu_f L} \]  

where \( \tau_R \) is the particle response time and \( \tau_F \) is a time characteristic of fluid motion; \( \rho_p \) is the density of the particle; \( d_p \) is the diameter of the particle; \( \mu_f \) is the dynamic viscosity of the fluid; \( L \) is the characteristic length associated with the fluid; \( U \) is the velocity magnitude of the fluid. It gives a measure of how quickly the particle is able to respond to changes in the flow field. A low value of Stokes number (\( St << 1 \)) suggests that the particle is able to follow streamlines and can maintain equilibrium with the carrier fluid, whereas a large Stoke number value (\( St >> 1 \)) indicates that particle has high inertia and will therefore break out of streamlines and consequently will be unaffected by turbulence in the flow field.

Discussion on the forces mentioned in Equation 1.17 is presented in the following sections for both spherical and non-spherical particles. Rotational dynamics of the particle is
insignificant if it is spherical. A discussion on the rotational dynamics of non-spherical particles has been omitted in this chapter (see Chapter-2, Section 2.1.3).

1.3.3.1 Pressure Gradient and Buoyancy Force

An existence of pressure gradient across the diameter of a particle gives rise to a net force acting on the particle. The net pressure force acting on a spherical particle is given as,

\[ \vec{F}_P = \int_S -p\vec{n}dS = \int_V -\Delta p dV \]  \hspace{1cm} (1.19)

where \( S \) denotes the surface of control volume and \( V \) denotes its volume. If it is assumed that the pressure gradient is constant over the volume of the particle, then,

\[ \vec{F}_p = -\Delta p V_p \]  \hspace{1cm} (1.20)

Pressure gradient due to buoyancy is given by,

\[ \Delta p = -\rho_f g \]  \hspace{1cm} (1.21)

where \( g \) is the acceleration due to gravity. Crowe et al. (2011) showed that for particles with high material densities both the pressure gradient and buoyancy forces will be negligible. However for bubbly flow the buoyant force is the most significant of all forces.

1.3.3.2 Unsteady Forces: Virtual Mass and Basset Force

The virtual mass and Basset force were first accounted for in the famous Basset-Boussinesq-Oseen (BBO) equation (Basset, 1888; Boussinesq, 1885; Oseen, 1928), applied to spherical particle moving at very low Reynolds number. A particle accelerating through a fluid has
to do additional work in order to also accelerate the fluid surrounding it. This results in an apparent increase in the inertia of the particle known as the ‘added mass’ or ‘virtual mass’. This force only comes into play when the particle is accelerating with respect to the fluid and hence is classified as an unsteady force. A derivation of the virtual mass force has been presented in Crowe et al. (2011). It is given as,

$$\vec{F}_{Vm} = \frac{\rho_f V_p^2}{2} (\vec{a}_f - \vec{a}_p) \quad (1.22)$$

where $\vec{a}_f$ is the acceleration of the fluid and $\vec{a}_p$ is the acceleration of the particle.

Basset force also called the ”history” force term comes into play due to the delay caused in boundary layer development when a particle is accelerated with respect to the fluid. For a sphere at low Reynolds number the Basset force is given by the relation (Basset, 1888)

$$\vec{F}_h = \frac{3}{2} D^2 \sqrt{\frac{\pi \rho_p \mu_f}{\rho_f}} \int^t_0 \frac{\vec{a}_f - \vec{a}_p}{\sqrt{t-t'}} dt' \quad (1.23)$$

According to the calculations performed by Hjelmfelt Jr and Mockros (1966) on both virtual mass and Basset forces, they concluded that both unsteady forces are insignificant for particles where $\frac{\rho_f}{\rho_p} < 10^{-3}$. Since the particulate flows in most applications have density ratios higher than $10^{-3}$, the unsteady forces are often neglected in simulations.

1.3.3.3 Drag Force

1.3.3.3.1 Drag Models for spherical particles Hydrodynamic drag acting on the particle due to the surrounding fluid has been thoroughly researched in literature for both spherical and non-spherical particles. For spherical particles the drag force is given
by the relation Crowe et al. (2011),

\[ \vec{F}_D = \frac{1}{2} \rho_f C_D A |\vec{u}_f - \vec{u}_p| (\vec{u}_f - \vec{u}_p) \]  \hspace{1cm} (1.24)

where \( A \) is the projected area of the particle in a direction perpendicular to relative fluid velocity \( \vec{u}_f - \vec{u}_p \). The value of drag coefficient depends on the shape and orientation of the particle and the Reynolds number. A relation for \( C_d \) in the Stokes flow regime \((Re < 1)\), i.e. creeping flow, was first derived by Stokes (1851) as,

\[ C_D = \frac{24}{Re_r} \]  \hspace{1cm} (1.25)

where \( Re_r \) is the Reynolds number of the surrounding fluid based on relative velocity. Oseen (1911) extended the analysis to include inertial effects such that,

\[ C_D = \frac{24}{Re_r} \left( 1 + \frac{3}{16} Re_r \right) \]  \hspace{1cm} (1.26)

which was valid up to Reynolds number \( \sim 5 \). Beyond 100 the flow behind the sphere starts to separate to form vortices behind it. This region is characterized by low pressure which further increases the drag experienced by the particle. Several correlations exist in literature for high Reynolds number flows (Schiller and Naumann, 1933; Putnam, 1961; Crowe et al., 1963; Clift and Gauvin, 1971) which were each valid for a particular range. One of the more recent and widely used correlation was given by Wen and Yu (1966) as

\[ C_D = \begin{cases} 
\frac{24(1+0.5Re_r^{0.687})}{Re_r} & \text{if } Re_r < 1000 \\
0.44 & \text{if } Re_r \geq 1000
\end{cases} \]  \hspace{1cm} (1.27)
An alternative method for calculating the drag force on any particle is by Lattice-Boltzmann simulations (see Van der Hoef et al. (2005)).

1.3.3.3.2 Drag Models for non-spherical particles

1. Brenner (1964):

The first drag relations for non-spherical particles were devised for ellipsoids by Brenner (1964) under creeping flow conditions. The drag force was given by,

\[ \vec{F}_D = \mu_f \pi a_p K (\vec{u}_f - \vec{u}_p) \] (1.28)

where \( a_p \) is the semi major axis of the ellipsoid. \( K \) is the resistance tensor in the global frame of reference depending on the orientation of the particle in the flow,

\[ K = M^{-1} K' M \] (1.29)

where \( M \) is the transformation matrix that transforms coordinates from body fitted coordinate system to global coordinate system (see Section 2.1.1). \( K' \) is the resistance tensor in the body fitted coordinate system, which is a diagonal matrix (Fan and Ahmadi, 1995),

\[ K' = \begin{bmatrix} K'_{xx} & 0 & 0 \\ 0 & K'_{yy} & 0 \\ 0 & 0 & K'_{zz} \end{bmatrix} \] (1.30)
where,

\[
K'_{xx} = \frac{8(\lambda^2 - 1)^{3/2}}{[(2\lambda^2 - 1)\ln(\lambda + \sqrt{\lambda^2 - 1}) - \lambda(\sqrt{\lambda^2 - 1})]} \tag{1.31}
\]

\[
K'_{yy} = K'_{zz} = \frac{16(\lambda^2 - 1)^{3/2}}{[(2\lambda^2 - 3)\ln(\lambda + \sqrt{\lambda^2 - 1}) + \lambda(\sqrt{\lambda^2 - 1})]} \tag{1.32}
\]

where \(\lambda\) is the aspect ratio of ellipsoid.

2. **Loth (2008):**

Loth (2008) studied the drag of both spherical and non-spherical particles in detail and provided corrections to the Stokesian drag for spherical particles for representing drag on non-spherical particles. Considering \(E\) to be the sphericity of the non-spherical particle, where \(E\) is defined as

\[
E = \frac{b}{a} \tag{1.33}
\]

where \(b\) is the semi-minor length and \(a\) is the semi-major length of spheroid. Note that spheroids have a single aspect ratio \(E\), whereas ellipsoids are characterized by three aspect ratios \((E_1, E_2, E_3)\). For \(E < 1\) the spheroid is called oblate and for \(E > 1\) it is termed as prolate. The correction factor with respect to Stokesian drag for creeping flow was given by,

\[
f_E = \frac{F_D(E, Re_p \to 0)}{F_{Dsphere}(d_p, Re_p \to 0)} \tag{1.34}
\]

\[
f_E = \frac{F_D(E, Re_p \to 0)}{3\pi d_p \mu f u_p} \tag{1.35}
\]

where \(F_D\) is corrected drag for the ellipsoid, \(Re_p \to 0\) signifies that the relation is
valid for creeping flow. Note that in the above relations the diameter of the sphere is such that its volume is same as the considered spheroid. For ellipsoids undergoing Stokesian flow the correction factor was given by,

\[
\frac{3}{<f_E>} = \frac{1}{f_{E1}} + \frac{1}{f_{E2}} + \frac{1}{f_{E3}}
\]  

(1.36)

Moving to high Reynolds number flows \(10^3 < Re_p < 10^5\) Loth (2008) defined, for convenience, correction factor as a ratio of the drag coefficients,

\[
C_{shape} = \frac{C_{D,shape}}{C_{D,sphere}}
\]

(1.37)

Calculations by and Jayaweera and Mason (1965) indicated that the drag coefficient for prolate particle with a circular cross-section, was a function of the projected area of the non-spherical particle perpendicular to the direction of the flow. Thus,

\[
C_D \approx A_{proj}^* \text{ for prolate particles}
\]

(1.38)

A relation for prolate ellipsoids was also derived in terms of surface area by Christiansen and Barker (1965),

\[
C_{shape} \approx 1 + 0.7 \left( A_{surf}^* - 1 \right)^{1/2} + 2.4 \left( A_{surf}^* - 1 \right) \text{ for prolate particles}
\]

(1.39)

For particles that are oblate or have distinctly non-circular cross-section the drag depended on the surface area such that the correction factor was given by,

\[
C_{shape} \approx 1 + 1.5 \left( A_{surf}^* - 1 \right)^{1/2} + 6.7 \left( A_{surf}^* - 1 \right) \text{ for oblate particles}
\]

(1.40)

Hölzer and Sommerfeld (2008) derived a relation for drag coefficient in terms of Reynolds number of the particle.

\[
C_D = \frac{8}{Re_p} \frac{1}{\sqrt{\phi_{\parallel}}} + \frac{16}{Re_p} \frac{1}{\sqrt{\phi}} + \frac{3}{\sqrt{Re_p}} \frac{1}{\phi^{3/4}} + 0.4210^{0.4(-\log\phi)^{0.2}} \frac{1}{\phi_{\perp}}
\]  

(1.41)

where \(\phi_{\parallel}\) is the lengthwise sphericity and \(\phi_{\perp}\) is the crosswise sphericity. They defined sphericity as the ratio between the surface area of the volume equivalent sphere and that of the considered particle. The above relation was found using the experimental data on non-spherical particles and was valid upto the critical Reynolds number.


Rosendahl (2000) suggested that for non-spherical particles the drag coefficient varies between two limits corresponding to the limiting shapes of the ellipsoid. The base drag coefficient corresponds to 0\(^o\) angle of incidence and the top drag coefficient corresponds to 90\(^o\). Thus, the drag coefficient at any angle of incidence \(\alpha\) is given by

\[
C_D(\alpha) = C_D(0^o) + (C_D(90^o) - C_D(0^o)) \sin^3 \alpha
\]

(1.42)

where the base drag coefficient is a function of the surface and projected areas,

\[
C_D(0^o) = C_D(90^o) \frac{A_{surf}}{A_{proj,\alpha=0^o}} \frac{R_\beta}{0.3678 \beta}
\]

(1.43)

where \(\beta\) is the ratio of semi-minor to semi-major axis. The projected area was defined
as a function of incidence angle,

\[ a_{proj} = \pi a^2 (\cos^2 \alpha + R_\beta \sin^2 \alpha) \tag{1.44} \]

where,

\[ R_\beta = \frac{A_{proj,\alpha=0}}{A_{proj,\alpha=0.5\pi}} \tag{1.45} \]

### 1.3.3.4 Lift Force

#### 1.3.3.4.1 Lift Force on spherical particles

There are two kinds of lift forces that may be experienced by a spherical particle viz., the Saffman and Magnus lift force. If a velocity gradient exist across the surface of particle, it induces rotation which in turn cause a pressure gradient. This pressure differential across the diameter of particle is responsible for the Saffman lift force. For low Reynolds number the expression for the force is (Saffman, 1965)

\[ \vec{F}_{Saff} = 1.61d_p^2(\mu \rho_f)^{0.5}|\vec{\omega}_f|^{-0.5} [ |\vec{u}_f - \vec{u}_p| \times \vec{\omega}_f ] \tag{1.46} \]

where

\[ \vec{\omega}_f = \Delta \times \vec{u}_f \tag{1.47} \]

Based on empirical results Dandy and Dwyer (1990) proposed a relation which pro-
vided a good fit for higher Reynolds number,

\[
\frac{F_L}{F_{Saff}} = \left\{ \begin{array}{ll}
(1 - 0.3314\beta^{0.5})e^{\left(-\frac{Re_p}{10}\right)} + 0.3314\beta^{0.5} & Re_p \leq 40 \\
0.0524(\beta Re_p)^{0.5} & Re_p > 40
\end{array} \right.
\]

(1.48)

where

\[
\beta = \frac{d_p}{2|\vec{u}_f - \vec{u}_p||\vec{\omega}_f|}
\]

(1.49)

Magnus lift in turn is caused by the rotation of the particle which induces a pressure gradient due to different fluid velocities on the both sides of the particle. Note that rotation may be caused by sources other than the velocity difference in contrast to the Saffman force. The Magnus force is given as,

\[
\vec{F}_{Mag} = \frac{1}{2}\rho_f|\vec{u}_f - \vec{u}_p|C_{LR}A\left(\frac{(\vec{u}_f - \vec{u}_p) \times \vec{\omega}_r}{|\vec{\omega}_r|}\right)
\]

(1.50)

where \(C_{LR}\) is the lift coefficient due to rotation and \(\vec{\omega}_r\) is the relative velocity of the fluid,

\[
\vec{\omega}_r = \vec{\omega}_p - \frac{1}{2}\nabla \times \vec{u}_f
\]

(1.51)

\(C_{LR}\) has been derived empirically in studies by Maccoll (1928), Barkla and Auchterlonie (1971) and Tanaka et al. (1990).

### 1.3.3.4.2 Force on non-spherical particles

1. **Hoerner (1965):**

   Lift force is often an insignificant contributor to the trajectory of spherical particles. However for non-spherical particles lift is substantial and can not be disregarded.
Nevertheless research on lift for non-spherical particles is limited as compared to drag force. Hoerner (1965) was the first to propose a relation for lift proportional to the drag given by the "cross-flow" principle,

\[ \frac{C_L}{C_D} = \sin^2 \alpha \cos \alpha \] (1.52)

where \( \alpha \) is the angle of incidence of the particle given by the angle between the major axis and the flow direction.


Mandø and Rosendahl (2010) argued that the relation by Hoerner (1965) provided a good fit to the experimental data in the Newton law regime. For low Reynolds number the ratio \( \frac{C_L}{C_D} \) deviated from the data. They proposed the following relation which asymptoted with the data for \( 30 < Re < 1500 \),

\[ \frac{C_L}{C_D} = \frac{\sin^2 \alpha \cos \alpha}{0.65 + 40Re_p^{0.72}} \] (1.53)

1.3.3.5 Hydrodynamic Torque

Torque acting on spherical particles due to the surrounding fluid is, for obvious reasons, insignificant. Only a few studies have explored the torque acting on the surface of spherical particle which arises mainly due to the shear stress distribution. For low Reynolds number (Happel and Brenner, 2012)

\[ \vec{T} = \pi \mu D^3 \left( \frac{1}{2} \Delta \times \vec{u} - \vec{\omega}_p \right) \] (1.54)
Rubinow and Keller (1961) gave the torque relation in terms of a torque coefficient \( (C_R) \) as,

\[
\vec{T} = \frac{\rho}{2} \left( \frac{d_p}{2} \right)^5 C_R |\vec{\omega}_p| |\vec{\omega}_p|
\]  

(1.55)

(1.56)

For higher Reynolds number, Dennis et al. (1980) studied the torque required to rotate a sphere in a viscous fluid. They extended the relation put forth by Rubinow and Keller (1961) to account for higher \( Re \). Thus \( C_R \) was given by,

\[
C_R = \begin{cases} 
\frac{64\pi}{Re} & \text{if } Re < 32 \\
\frac{12.9}{Re^{0.8}} + \frac{128.4}{Re} & \text{if } 32 < Re < 100
\end{cases}
\]  

(1.57)

For non-spherical particles, owing to the anisotropic shape, the hydrodynamic torque is significant. The orientation of the particle with respect to the surrounding fluid also effects drag and lift force acting on the particle, as discussed previously, increasing the need for accurate torque models. Jeffery (1922) was the first to give torques correlations for ellipsoids under Stokes flow conditions.

\[
T_x = \frac{32\pi \mu f a^3 \lambda}{3(\alpha_2 + \alpha_3)} (\Omega_{xy} - \omega_x)
\]  

(1.58)

\[
T_y = \frac{16\pi \mu f a^3 \lambda}{3(\alpha_3 + \lambda^2 \alpha_1)} \left[ (1 - \lambda^2) S_{xz} + (1 + \lambda^2)(\Omega_{xz} - \omega_y) \right]
\]  

(1.59)

\[
T_z = \frac{16\pi \mu f a^3 \lambda}{3(\alpha_2 + \lambda^2 \alpha_1)} \left[ (\lambda^2 - 1) S_{yx} + (1 + \lambda^2)(\Omega_{yx} - \omega_z) \right]
\]  

(1.60)

(1.61)
where, $S$ is the strain rate tensor of fluid,

$$ S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) $$ (1.62)

$\Omega$ is the rotation tensor of the fluid,

$$ \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) $$ (1.63)

$\alpha_i$ are defined as,

$$ \alpha_1 = -\frac{2}{\lambda^2 - 1} - \frac{\lambda}{(\lambda^2 - 1)^{3/2}} \ln \left[ \frac{\lambda - (\lambda^2 - 1)^{1/2}}{\lambda + (\lambda^2 - 1)^{1/2}} \right] $$ (1.64)

$$ \lambda_2 = \lambda_3 = \frac{2}{\lambda^2 - 1} - \frac{\lambda}{(\lambda^2 - 1)^{3/2}} \ln \left[ \frac{\lambda - (\lambda^2 - 1)^{1/2}}{\lambda + (\lambda^2 - 1)^{1/2}} \right] $$ (1.65)

where $\lambda$ is the aspect ratio of the ellipsoid. The above relation is used most widely in literature for hydrodynamic torque on non-spherical particles even though Jeffery (1922) devised it for low Reynolds number only. Other than the empirical and analytical correlations drag, lift and torque coefficients can be determined for specific particle shapes using Lattice Boltzmann simulations (Hölzer and Sommerfeld, 2009) or true DNS simulations (Zastawny et al., 2012).

### 1.4 Chapter Summary

This chapter introduces the basic concepts related to modelling the transport of spherical and non-spherical particles in Eulerian flow fields. Collision models that exist in literature have been introduced and a critical discussion of multi-Sphere (MS) method, the current academic and industrial standard for non-spherical particle modelling, has been
presented. From published studies that analyzed the validity of the MS approach it can be concluded that it does not provide a reliable collision modelling technique. An increase in the number of comprising spheres does not translate into a more accurate depiction of actual non-spherical particles. This provides the motivation for the present work for the development of a novel collision model.

Several existing correlations for calculating hydrodynamic forces and torques on particles suspended in a flow field have been presented. Note that drag, lift and torque relations specifically used in this study have been omitted in this chapter but are discussed at length in Chapter 2.
Chapter 2

Dynamics of Non-Spherical Particles

In comparison to spherical particles the dynamics of non-spherical particles are a lot more complicated to model. Anisotropic shapes of non-spherical particles makes it imperative to track the varying rotational orientation of particles. Specifically, the effective cross sectional area of particle relative to the local fluid velocity keeps changing with time, which in turn affects the direction and magnitude of the drag and lift forces acting on it. Moreover, it is also necessary to take into account the torques acting on the particle due to the surrounding flow field. Adding further to the complication is the collision mechanics of the non-spherical particles. Collision detection of spherical particles may be easily calculated based on the condition:

\[ l \leq r_1 + r_2 \]  \hspace{1cm} (2.1)

where \( l \) is the distance between the centers of particles, while \( r_1 \) and \( r_2 \) are the particle radii. In contrast, detecting collision events deterministically for non-spherical
particles is very complex. For example, calculating the collision of ellipses involves the solution of a fourth-order system of equations which represents a family of overlapping ellipses. To circumvent this problem the non-spherical particle may be modelled as a system of several spherical particles which would then reduce the process of collision of two non-spherical particles to collision detection among several spherical particles representing each system. This method referred to as the Multi-Sphere approach (Favier et al., 1999; Jensen et al., 1999) is popularly used in commercial CFD packages such as EDEM, Star-CCM+ and PFC. Markauskas et al. (2010) have investigated the adequacy of Multi-Sphere approximation for simulating ellipsoids. Although this method lowers the mathematical complexity of contact detection and modelling collision forces, representing a single ellipsoid by a collection of spheres increases the computational expense by a significant factor (Markauskas et al., 2010).

Elliptical shapes are the subject of most non-spherical particle studies because in most engineering applications, including granular material (Markauskas et al., 2010; Rothenburg and Bathurst, 1991) and inhaled aerosols, ellipses are capable of representing the real particles most closely. Therefore the collision models developed in this thesis are restricted to ellipses. A novel contact detection and collision modelling approach for ellipses has been introduced in the latter part of this chapter which takes into account the actual equations of ellipses for contact detection but models the collision forces by a method which is akin to the existing Multi-Sphere approach; however, it does not require that the ellipse be represented by a collection of spheres. Moreover the model has been generalized to include the case where the ellipses have a rough surface. More general concepts which are pertinent to the transport of anisotropic particles in an Eulerian field flow are introduced first.
2.1 Lagrangian Phase Modelling

2.1.1 Coordinate Transformation

For the purpose of collision dynamics and also the evaluation of hydrodynamic forces and torques it is convenient to define a coordinate system which is fixed and aligned to the principal axes of the body. A map between the coordinates in the two frames is given as

\[ \mathbf{x} = M \mathbf{x}' + \mathbf{x}_p \]  

(2.2)

where \( \mathbf{x} \) and \( \mathbf{x}' \) represent the coordinates in the global frame and body frame respectively. \( \mathbf{x}_p \) represents the position of the origin of the body coordinate system with respect to the origin of the global coordinate system. \( M \) is the transformation matrix that maps the two coordinate systems. Coordinate transformation may be achieved by either Euler angles or quaternions. Quaternions are, however, increasingly being used to represent rotation due to the absence of singularity problems (Evans and Murad, 1977) which may arise in the case of Euler angles when they approach ±90°. Since quaternions are represented by half-angles, singularities can never occur while dealing with them. Note that these problems may not necessarily arise in planar problems which are the subject of this work, nevertheless, quaternions are used for modelling rotations in the present code so that it may be easier to convert the code to 3D in the future.

Transformation using Euler angles is achieved by three successive rotations about the original principal axes (Henderson, 1977). For instance, the final coordinate position may be reached by successive rotations about the X, Y and Z axes, which would lead to the following transformation matrix
\[
M = \begin{pmatrix}
\cos\theta_2 \cos\theta_3 & -\cos\theta_2 \sin\theta_3 & \sin\theta_2 \\
\cos\theta_1 \sin\theta_3 + \sin\theta_1 \sin\theta_2 \cos\theta_3 & \cos\theta_1 \cos\theta_3 - \sin\theta_1 \sin\theta_2 \sin\theta_3 & -\sin\theta_1 \cos\theta_2 \\
\sin\theta_1 \sin\theta_3 - \cos\theta_1 \sin\theta_2 \cos\theta_3 & \sin\theta_1 \cos\theta_3 + \cos\theta_1 \sin\theta_2 \sin\theta_3 & \cos\theta_1 \cos\theta_2
\end{pmatrix}
\]

(2.3)

Note that the sequence of rotation in which the transformation is achieved is insignificant as long as the same sequence is followed for all transformations.

Quaternions, first introduced by Hamilton (1844), are a number system that extends the complex numbers into a 4 dimensional space consisting of a scalar and a vector part. A quaternion is defined by

\[
q = [q_1, q_2, q_3, q_4]
\]

(2.4)

Rotational transformations are performed using unit quaternions i.e. \(|q| = 1\). Thus the four coefficients must satisfy the relation

\[
\sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} = 1
\]

(2.5)

Rotation through a quaternion is achieved by a single rotation of \(\omega\) about a vector with the direction cosines \(\alpha, \beta, \text{ and } \gamma\) with respect to the axes of the original coordinate system. Thus,
\[ q = e^{\frac{\omega}{2}(\cos \alpha i + \cos \beta j + \cos \gamma k)} \]  

(2.6)

\[ \Rightarrow q_1 = \cos \frac{\omega}{2} \]  

(2.7)

\[ \Rightarrow q_2 = \cos \alpha \sin \frac{\omega}{2} \]

\[ \Rightarrow q_3 = \cos \beta \sin \frac{\omega}{2} \]

\[ \Rightarrow q_4 = \cos \gamma \sin \frac{\omega}{2} \]

The resulting transformation matrix in terms of quaternions is,

\[
M = \begin{pmatrix}
q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 + q_1q_3) \\
2(q_2q_3 + q_1q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3q_4 - q_1q_2) \\
2(q_2q_4 - q_1q_3) & 2(q_3q_4 + q_1q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2
\end{pmatrix}
\]

(2.8)

Comparing Equations (2.3) and (2.8) the quaternions may be derived in terms of the Euler angles (using trigonometric half-angle identities),

\[
q_1 = -\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} + \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2}
\]

(2.9)

\[
q_2 = \sin \frac{\theta_1}{2} \cos \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_2}{2} \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2}
\]

\[
q_3 = -\sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2}
\]

\[
q_4 = \sin \frac{\theta_1}{2} \sin \frac{\theta_2}{2} \cos \frac{\theta_3}{2} + \sin \frac{\theta_3}{2} \cos \frac{\theta_1}{2} \cos \frac{\theta_2}{2}
\]

In the present work with planar transformations, the Euler angles \( \theta_1 \) and \( \theta_2 \) will be
zero. Consequently the coefficients \( q_2 \) and \( q_3 \) will remain inactive.

### 2.1.2 Translational Dynamics

This section details the relation specifically used in this work for spherical and non-spherical transport. In the Lagrangian phase the particles are treated as discrete and their dynamics are described by Newton’s equations of motion. The general equation of motion for translation is:

\[
m_p \frac{d\vec{v}_p}{dt} = \vec{F}_D + \vec{F}_P + \vec{F}_{VM} + \vec{F}_g + \vec{F}_L + \vec{F}_{pp} + \vec{F}_{pw} + \vec{F}_h
\]  

(2.10)

where \( \vec{F}_D \) is the drag force; \( \vec{F}_P \) is the force due to the pressure gradient; \( \vec{F}_{VM} \) is the virtual mass force; \( \vec{F}_g \) is the net gravitational force; \( \vec{F}_L \) is the lift force; \( \vec{F}_{pp} \) is the force due to interaction of particles; \( \vec{F}_{pw} \) is the force due to interaction of a particle with boundary and \( \vec{F}_h \) is the Basset force due to history effects. For particle-laden flows where \( \rho_f/\rho_p \sim 10^{-3} \) most of the forces mentioned above are negligible (Crowe et al., 1996).

Thus Equation (2.10) can be reduced to,

\[
m_p \frac{d\vec{v}_p}{dt} = \vec{F}_D + \vec{F}_g + \vec{F}_L + \vec{F}_{pp} + \vec{F}_{pw}
\]

(2.11)

The drag and lift forces are represented by the general relations (van Wachem et al.,
where $C_D$ and $C_L$ are the drag and lift coefficients respectively, $\vec{v}_r = \vec{v}_f - \vec{v}_p$ is the relative velocity of the fluid with respect to the particle at the particle position and $d_p^2$ is the equivalent spherical diameter of the particle, i.e., the diameter of the sphere with the same volume as the original particle (van Wachem et al., 2015). The drag and lift coefficients are a function of the particle shape and orientation. For spherical particles a relation for the drag coefficient $C_D$ as employed by Vreman et al. (2009) is used,

\[
C_D = \begin{cases} 
24[1+0.5Re_p^{0.687}] & \text{if } Re_p < 1000 \\
0.44 & \text{if } Re_p \geq 1000 
\end{cases} 
\]  

(2.14)

$Re_p$ is the particle Reynolds number and is defined as:

\[
Re_p = \frac{\rho f |\vec{v}_r| d_p}{\mu} 
\]  

(2.15)

Note that Mallouppas and van Wachem (2013) have included volume fraction of the particle in the drag relation which is required to calculate the two-way coupling term. However, in this study all simulations considered are one-way coupled, therefore the use of the volume fraction term is omitted.

Zastawny et al. (2012) performed DNS simulations on non-spherical particles with different aspect ratios. Their study demonstrated that the translational dynamics of the
non-spherical particles may be demonstrated by the dynamics of spheres with equivalent
diameter. The drag and lift coefficients, however, are a function of the orientation of the
particle with respect to local fluid velocity, i.e.,

\[ C_D(\phi) = C_{D,\phi=0^0} + (C_{D,\phi=90^0} - C_{D,\phi=0^0})(\sin\phi)^{a_0} \]  \hspace{1cm} (2.16)

Specifically,

\[ C_{D,\phi=0^0} = \frac{a_1}{Re_p^{a_2}} + \frac{a_3}{Re_p^{a_4}} \]  \hspace{1cm} (2.17)

\[ C_{D,\phi=90^0} = \frac{a_5}{Re_p^{a_6}} + \frac{a_7}{Re_p^{a_8}} \]  \hspace{1cm} (2.18)

\[ C_L(\phi) = \frac{b_1}{Re_p^{b_2}} + \frac{b_3}{Re_p^{b_4}}(\sin\phi)^{b_5+b_6Re_p^{b_7}}(\cos\phi)^{b_8+b_9Re_p^{b_{10}}} \]  \hspace{1cm} (2.19)

Here \( \phi \) is the angle of incidence of the particle i.e. the angle made by the principal
axis of the particle with the local fluid velocity in the body coordinate system (see Figure
2.1). Thus

\[ \phi = \left| \arctan \left( \frac{v_{ry}^b}{v_{rx}^b} \right) \right| \]  \hspace{1cm} (2.20)

where \( v_{rx}^b \) and \( v_{ry}^b \) are the projections of the local relative fluid velocity along the
minor and major axis of the particle respectively. The superscript indicates that the
velocities are in the body coordinate system. \( a_i \) and \( b_i \) are the coefficients that were
calculated by Zastawny et al. (2012) for different particle shapes. They are enumerated in Table 2.1.

\[ F_D^b = \frac{1}{2} \rho \frac{1}{4} \pi d_p^2 C_D |\vec{v}_r^b| |\vec{v}_r^b| \] (2.21)

\[ F_{L,x}^b = \frac{1}{2} \rho \frac{1}{4} \pi d_p^2 C_L |\vec{v}_r^b|^2 \sin \phi \text{sign}(v_{r,x}^b) \] (2.22)

\[ F_{L,y}^b = \frac{1}{2} \rho \frac{1}{4} \pi d_p^2 C_L |\vec{v}_r^b|^2 \cos \phi \text{sign}(v_{r,y}^b) \] (2.23)

Note that due to the high Stokes number of the particles considered in this work, the Saffman and Magnus lift forces acting on the particle are neglected (van Wachem et al., 2015).
2.1.3 Rotational Dynamics

The equations of motion for the rotational speed and orientation of a particle are too elaborate to express in the global coordinate system. Thus, to describe the rotation of the particle the equations are first solved in the body fitted coordinate system and then converted to the global frame. The equations of particle rotation are:

\[ I^b_x \frac{d\omega^b_x}{dt} = \sum T_x^b + \omega^b_y \omega^b_z (I^b_y - I^b_z) \] (2.24)
\[ I^b_y \frac{d\omega^b_y}{dt} = \sum T_y^b + \omega^b_x \omega^b_z (I^b_z - I^b_x) \] (2.25)
\[ I^b_z \frac{d\omega^b_z}{dt} = \sum T_z^b + \omega^b_y \omega^b_x (I^b_x - I^b_y) \] (2.26)

where \( I^b_x, I^b_y \) and \( I^b_z \) are the principal moments of inertia of the particle along the x, y and z axes of the particle fitted coordinate system. \( \omega^b_i \) represents the angular velocity of the particle in the body frame. The hydrodynamic torques acting on the particle arises due to two reasons (van Wachem et al., 2015). First is the resistance offered by the surrounding fluid to the angular motion of the particle, termed the resisting torque, defined as

\[ T_{x,R}^b = -C_R \frac{d\rho}{d\eta} \frac{\Omega_{x}^b}{2} \Omega_{x}^b \] (2.27)
\[ T_{y,R}^b = -C_R \frac{d\rho}{d\eta} \frac{\Omega_{y}^b}{2} \Omega_{y}^b \] (2.28)
\[ T_{z,R}^b = -C_R \frac{d\rho}{d\eta} \frac{\Omega_{z}^b}{2} \Omega_{z}^b \] (2.29)

In the above equations the \( \Omega^b \) terms are the components of the angular velocity of
the fluid surrounding the particle with respect to the angular velocity of the particle in
the body frame. Thus
\[
\vec{\Omega}_b = \frac{1}{2} \nabla \times \vec{v}_r^b - \vec{\omega}_p^b
\]  
(2.30)

where \( \nabla \times \vec{v}_r^b \) is the local relative vorticity of the fluid and \( \vec{\omega}_p^b \) is the angular
velocity of the particle in the body frame. \( C_R \) is the coefficient of resisting torque derived
by Zastawny et al. (2012)

\[
C_R = r_1Re_R^{r_2} + \frac{r_3}{Re_R}^{r_4}
\]  
(2.31)

where \( Re_R \) is the rotational particle Reynolds number,

\[
Re_R = \frac{\rho_1 d_p^2 |\vec{\Omega}|}{\mu}
\]  
(2.32)

The second torque acting on the particle is the pitching moment exerted by the fluid
due to the anisotropic shape of the particle. As explained by van Wachem et al. (2015),
for axisymmetric particles with \( x_b \) as the principal axis the pitching torque about the
x-axis will be zero. The other two components are given by

\[
T_{y,P}^b = \frac{1}{4} \rho \frac{\pi}{d}^2 d_p C_T \frac{v_{r,y}^b}{\sqrt{v_{r,y}^2 + v_{r,z}^2}} \text{sign}(v_{r,x} b_{r,z}^b)
\]  
(2.33)

\[
T_{z,P}^b = \frac{1}{4} \rho \frac{\pi}{d}^2 d_p C_T \frac{v_{r,y}^b}{\sqrt{v_{r,y}^2 + v_{r,z}^2}} \text{sign}(v_{r,x} b_{r,y}^b)
\]  
(2.34)

where \( C_T \) is the pitching torque coefficient defined by

\[
C_T(\phi) = \left( \frac{c_1}{Re} + \frac{c_3}{Re^4} \right) \sin(\phi)^{c_5+c_6Re^{c_7}} \cos(\phi)^{c_8+c_9Re^{c_{10}}}
\]  
(2.35)
All the coefficient terms for different particle shapes that are considered in the present work are listed in Table 2.1

### 2.2 Numerical Integration of Particle Velocity and Position

The translational equations of motion are integrated using Verlet scheme expansion (van Wachem et al., 2015)

\[
\vec{x}_p(t + \Delta t) = 2\vec{x}_p(t) - \vec{x}_p(t - \Delta t) + \Delta t^2 \vec{a}_p(t) + O\Delta t^4 \quad (2.36)
\]

where \( \vec{a}_p \) is the acceleration experienced by the particle due to the forces acting on it as given by Equation (2.11). The velocity of the particle is given by a similar expansion

\[
\vec{v}_p(t + \Delta t) = \frac{\vec{x}_p(t + \Delta t) - \vec{x}_p(t - \Delta t)}{2\Delta t} + O\Delta t^2 \quad (2.37)
\]

For the rotational motion, since the particle orientation is represented by quaternions, the angular velocity and orientation are related by the equation

\[
\frac{dq}{dt} = \begin{pmatrix}
0 & -\omega_z & -\omega_y & 0 \\
\frac{\omega_y}{2} & 0 & \omega_x & \frac{-\omega_z}{2} \\
\frac{-\omega_x}{2} & -\omega_y & 0 & \omega_z \\
\frac{\omega_z}{2} & \frac{-\omega_x}{2} & -\omega_y & 0
\end{pmatrix}
\begin{pmatrix}
q_1 \\
q_2 \\
q_3 \\
q_4
\end{pmatrix} \quad (2.38)
\]

Equation (2.38) may also be updated using Taylor’s expansion (Evans and Murad,
Table 2.1: Values of coefficients used in the force and torque relations for different particle shapes (Zastawny et al., 2012)

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Ellipsoid 1</th>
<th>Ellipsoid 2</th>
<th>Disc</th>
<th>Fiber</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_0$</td>
<td>2</td>
<td>1.95</td>
<td>1.96</td>
<td>2.12</td>
</tr>
<tr>
<td>$a_1$</td>
<td>5.1</td>
<td>18.12</td>
<td>5.82</td>
<td>20.35</td>
</tr>
<tr>
<td>$a_2$</td>
<td>0.48</td>
<td>1.023</td>
<td>0.44</td>
<td>0.98</td>
</tr>
<tr>
<td>$a_3$</td>
<td>15.52</td>
<td>4.26</td>
<td>15.56</td>
<td>2.77</td>
</tr>
<tr>
<td>$a_4$</td>
<td>1.05</td>
<td>0.384</td>
<td>1.068</td>
<td>0.396</td>
</tr>
<tr>
<td>$a_5$</td>
<td>24.68</td>
<td>21.52</td>
<td>35.41</td>
<td>29.14</td>
</tr>
<tr>
<td>$a_6$</td>
<td>0.98</td>
<td>0.99</td>
<td>0.96</td>
<td>0.97</td>
</tr>
<tr>
<td>$a_7$</td>
<td>3.19</td>
<td>2.86</td>
<td>3.63</td>
<td>3.66</td>
</tr>
<tr>
<td>$a_8$</td>
<td>0.21</td>
<td>0.26</td>
<td>0.05</td>
<td>0.16</td>
</tr>
<tr>
<td>$b_1$</td>
<td>6.079</td>
<td>0.083</td>
<td>12.111</td>
<td>8.652</td>
</tr>
<tr>
<td>$b_2$</td>
<td>0.898</td>
<td>-0.21</td>
<td>1.036</td>
<td>0.815</td>
</tr>
<tr>
<td>$b_3$</td>
<td>0.704</td>
<td>1.582</td>
<td>3.887</td>
<td>0.407</td>
</tr>
<tr>
<td>$b_4$</td>
<td>-0.028</td>
<td>0.851</td>
<td>0.109</td>
<td>-0.197</td>
</tr>
<tr>
<td>$b_5$</td>
<td>1.067</td>
<td>1.842</td>
<td>0.812</td>
<td>0.978</td>
</tr>
<tr>
<td>$b_6$</td>
<td>0.0025</td>
<td>-0.802</td>
<td>0.249</td>
<td>0.036</td>
</tr>
<tr>
<td>$b_7$</td>
<td>0.818</td>
<td>-0.006</td>
<td>-0.198</td>
<td>0.451</td>
</tr>
<tr>
<td>$b_8$</td>
<td>1.049</td>
<td>0.874</td>
<td>5.821</td>
<td>1.359</td>
</tr>
<tr>
<td>$b_9$</td>
<td>0</td>
<td>0.009</td>
<td>-4.717</td>
<td>-0.43</td>
</tr>
<tr>
<td>$b_{10}$</td>
<td>0</td>
<td>0.57</td>
<td>0.007</td>
<td>0.007</td>
</tr>
<tr>
<td>$c_1$</td>
<td>2.078</td>
<td>0.935</td>
<td>3.782</td>
<td>0.011</td>
</tr>
<tr>
<td>$c_2$</td>
<td>0.279</td>
<td>0.146</td>
<td>0.237</td>
<td>-0.656</td>
</tr>
<tr>
<td>$c_3$</td>
<td>0.372</td>
<td>-0.469</td>
<td>2.351</td>
<td>8.909</td>
</tr>
<tr>
<td>$c_4$</td>
<td>0.018</td>
<td>0.145</td>
<td>0.236</td>
<td>0.396</td>
</tr>
<tr>
<td>$c_5$</td>
<td>0.98</td>
<td>0.116</td>
<td>-0.394</td>
<td>2.926</td>
</tr>
<tr>
<td>$c_6$</td>
<td>0</td>
<td>0.748</td>
<td>1.615</td>
<td>-1.28</td>
</tr>
<tr>
<td>$c_7$</td>
<td>0</td>
<td>0.041</td>
<td>-0.044</td>
<td>0.037</td>
</tr>
<tr>
<td>$c_8$</td>
<td>1</td>
<td>0.221</td>
<td>-0.537</td>
<td>-15.236</td>
</tr>
<tr>
<td>$c_9$</td>
<td>0</td>
<td>0.657</td>
<td>1.805</td>
<td>16.757</td>
</tr>
<tr>
<td>$c_{10}$</td>
<td>0</td>
<td>0.044</td>
<td>-0.037</td>
<td>-0.006</td>
</tr>
<tr>
<td>$r_1$</td>
<td>0.23</td>
<td>0.573</td>
<td>3.812</td>
<td>0.024</td>
</tr>
<tr>
<td>$r_2$</td>
<td>-0.116</td>
<td>-0.154</td>
<td>-0.13</td>
<td>0.168</td>
</tr>
<tr>
<td>$r_3$</td>
<td>96.378</td>
<td>116.61</td>
<td>283.03</td>
<td>77.314</td>
</tr>
<tr>
<td>$r_4$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
1977). However, the problem with this method is that the norm of the integrated value of quaternion may exceed unity. Whitmore (2000) introduced a method for integrating unit quaternions without the need for manually normalizing quaternions after every time step. Alternative algorithms were developed by Betsch and Siebert (2009), Karney (2007) and Sabatini (2005).

In the present work, a method introduced by Whitmore (2000), termed as the direct multiplication method, is used. Equation (2.38) can be expressed in tensor form as

\[
\dot{q} = \psi(\omega_x, \omega_y, \omega_z)q
\]  

(2.39)

Whitmore (2000) introduced an integrating factor i.e.,

\[
I(t) = \exp \left[ \int_{t_0}^{t} -\psi dt \right]
\]  

(2.40)

Using this to integrate Equation (2.39) from time \( t_0 \) to \( t \) we obtain,

\[
q(t) = -Iq_0
\]  

(2.41)

For a discrete time step, \( \delta t \) between time levels \( n \) and \( n + 1 \), the above equation becomes

\[
q_{n+1} = \exp \begin{pmatrix}
0 & -\frac{\omega_y}{2} & -\frac{\omega_z}{2} \\
\frac{\omega_y}{2} & 0 & \frac{\omega_z}{2} \\
\frac{\omega_z}{2} & -\frac{\omega_y}{2} & 0 \\
\end{pmatrix} \delta t q_n
\]  

(2.42)

Expanding the exponential term in the above equation with the Maclaurin series, we
may reduce the equation to a simplified form (Whitmore, 2000; Zhao and van Wachem, 2013b) which gives a convenient equation for quaternion at time level \( n + 1 \),

\[
\mathbf{q}_{n+1} = \begin{bmatrix}
    \cos \frac{||\mathbf{\omega}||}{2} 
    \mathbf{I} + 
    \frac{2}{||\mathbf{\omega}||} \sin \frac{||\mathbf{\omega}||}{2} 
    \begin{pmatrix}
        0 & -\frac{\omega_z}{2} & -\frac{\omega_y}{2} & -\frac{\omega_x}{2} \\
        -\frac{\omega_z}{2} & 0 & -\frac{\omega_x}{2} & -\frac{\omega_y}{2} \\
        -\frac{\omega_y}{2} & -\frac{\omega_x}{2} & 0 & -\frac{\omega_z}{2} \\
        -\frac{\omega_x}{2} & -\frac{\omega_y}{2} & -\frac{\omega_z}{2} & 0
    \end{pmatrix}
\end{bmatrix} \delta t \mathbf{q}_n
\] (2.43)

In the above equation \( \mathbf{I} \) is the fourth order identity matrix and \( ||\mathbf{\omega}|| \) is the magnitude of the angular velocity. The terms on the right hand side may be represented more conveniently by a unit quaternion,

\[
\mathbf{q}_{n+1} = \tilde{\mathbf{q}}_n \mathbf{q}_n
\] (2.44)

Zhao and van Wachem (2013b) enhanced the method described above by updating the quarternions over fractions of the selected time step. The angular velocity was integrated using a predictor-corrector method and the quarternions were integrated using the direct multiplication method described above. Hence the name predictor-corrector direct multiplication (PCDM) method. The angular velocity in body coordinates at time level \( n \) is given by

\[
\mathbf{\omega}_n^b = \mathbf{q}_n^{-1} \mathbf{\omega}_n \mathbf{q}_n
\] (2.45)

The angular velocity at quarter and half of time step is then updated using a basic Lie-Euler method (Allen and Tildesley, 1989)
\[
\omega_{n+\frac{1}{4}}^b = \omega_n^b + \frac{1}{4} \alpha_n^b \delta t
\]  
(2.46)

\[
\omega_{n+\frac{1}{2}}^b = \omega_n^b + \frac{1}{2} \alpha_n^b \delta t
\]  
(2.47)

where \(\alpha_n\) is the angular acceleration acting on the particle at time level \(n\). A prediction of the angular velocity in global coordinates at quarter step is then made using

\[
\omega_{n+\frac{1}{4}} = q_n \omega_{n+\frac{1}{4}}^b q_n^{-1}
\]  
(2.48)

The quaternion at half time step is now predicted using the angular velocity at at quarter time step. Thus using Equation (2.43) we get

\[
q_{n+\frac{1}{2}} = \left[\cos \frac{||\omega_{n+\frac{1}{2}}||}{2} I + \frac{2}{||\omega_{n+\frac{1}{2}}||} \sin \frac{||\omega_{n+\frac{1}{2}}||}{2} \left[\Omega_{n+\frac{1}{4}}\right]\right] \frac{\delta t}{2} q_n
\]  
(2.49)

where \(\left[\Omega_{n+\frac{1}{4}}\right]\) is the corresponding angular velocity matrix as suggested by Equation (2.43). The angular velocity in global coordinates at half time step is then calculated using the above predicted value of quaternion

\[
\omega_{n+\frac{1}{2}} = q_{n+\frac{1}{2}} \omega_{n+\frac{1}{4}}^b q_{n+\frac{1}{2}}^{-1}
\]  
(2.50)

The value of unit quaternion at time level \(n + 1\) is then corrected,

\[
q_{n+1} = \left[\cos \frac{||\omega_{n+\frac{1}{2}}||}{2} I + \frac{2}{||\omega_{n+\frac{1}{2}}||} \sin \frac{||\omega_{n+\frac{1}{2}}||}{2} \left[\Omega_{n+\frac{1}{4}}\right]\right] \delta t q_n
\]  
(2.51)

The angular velocity at time level \(n + 1\) is finally obtained using the latest value of
quaternion

\[ \omega_{n+1}^b = \omega_n^b + \alpha_n^b \delta t \]  \hspace{1cm} (2.52)

\[ \omega_{n+1} = q_{n+1} \omega_{n+1} b q_{n+1}^{-1} \]  \hspace{1cm} (2.53)

\section*{2.3 Collision Dynamics}

\subsection*{2.3.1 Contact Detection}

For the particle concentration considered in this work, it is imperative to identify the possible particle-particle collisions. Also, to keep the particles in the confines of the computational domain it is necessary to detect the possible particle-wall collisions.

Figure 2.2: Contact detection in multi-Sphere approach is reduced to finding overlap between comprising sub-spheres

Considering the particles in 2-D as ellipses the problem of contact detection of neigh-
boring particles may be approached in two different ways. Either as the multi-sphere approach (or the sphere intersection method as described by Langston et al. (2004)) (see Figure 2.2) or by describing the pair of particle surfaces by an overlapping function (Allen et al. (1989), Etayo et al. (2006), Hughes and Chraibi (2012), Džiugys and Peters (2001b)). The first method is more commonly used because of the mathematical simplicity of contact detection using Equation (2.10). Also, this method is also more general as it can be applied to any arbitrary particle shape by simply arranging spheres inside the volume of the particle. The method, however, requires that the particle be represented by a large number of spheres for increased accuracy. Moreover, the requirement of tracking all packed spheres increases the computational costs. The paper by van Wachem et al. (2015) served as the benchmark study for the present work in using this method for implementing particle-particle and particle-wall collision dynamics. However, in this work the second method is employed with the objective of eliminating the need for tracking multiple spheres, but at the same time using similar mechanics for calculation of collision forces and torques, as detailed in Section 2.3.3. Specifically, the collision dynamics modelled in this work is based on the soft-particle assumption, which allows actual physical overlap between the surfaces of the particle pair. The alternative approach, i.e., hard-particle assumption is modelled using event driven algorithm (Donev et al., 2005) and is more popularly used for molecular dynamics (MD) simulations. Details of the differences between the hard and soft particle modelling approaches were discussed in Chapter-1 (see Section 1.3.1.2.1).

2.3.1.1 Particle-Particle Overlap Detection

Wang et al. (2001) studied the characteristic equation of two ellipsoids and identified the relations for its roots which established the conditions for intersection of ellipsoids. Etayo
et al. (2006) adopted the results for ellipses and extended the analysis giving detailed relations for all possible relative positions of a pair of ellipses (see Table 2.2).

Consider the pair of ellipses being analyzed are represented by the indices \(i\) and \(j\). All collision equations are solved in the body coordinate system. Let \(x^i, y^i\) and \(x^j, y^j\) be the coordinate system fixed to the principal axes of the ellipses \(i\) and \(j\) respectively. Conversion of coordinates from the global coordinate system to body coordinate system and vice-versa may be carried out using equations detailed in section 2.1.1. Equation of the ellipses in their respective coordinates systems are

\[
\begin{align*}
x^i + \frac{y^i}{B_i^2} - 1 &= 0 \\
x^j + \frac{y^j}{B_j^2} - 1 &= 0
\end{align*}
\]

where \(A_i, B_i\) and \(A_j, B_j\) are the semi-major and semi-minor axes of the two ellipses. For further analysis it is required that the equations are mapped into a single coordinate system. Thus, the relation connecting the coordinates of particle \(j\) with \(i\) is

\[
x^j = M_j^{-1} (M^i x^i + x^i_{ip} - x^j_{ip})
\]

where \(M^i\) and \(M^j\) are the respective transformation matrices; \(x^i_{ip}\) and \(x^j_{ip}\) are the position vectors of centers of particle \(i\) and \(j\) with respect to the global coordinate system. Equation of both ellipses, in the coordinate system fixed to particle \(i\), may then be represented in the general implicit form as
Table 2.2: Relative position of two ellipses (Etayo et al., 2006)

<table>
<thead>
<tr>
<th>Description</th>
<th>Figure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Transversal at 4 points</td>
<td><img src="image1.png" alt="Figure 1" /></td>
</tr>
<tr>
<td>2. Transversal at 2 points</td>
<td><img src="image2.png" alt="Figure 2" /></td>
</tr>
<tr>
<td>3. Separated</td>
<td><img src="image3.png" alt="Figure 3" /></td>
</tr>
<tr>
<td>4. One contained in other</td>
<td><img src="image4.png" alt="Figure 4" /></td>
</tr>
<tr>
<td>5. Tranversal at 2 points and tangent at other point</td>
<td><img src="image5.png" alt="Figure 5" /></td>
</tr>
<tr>
<td>6. Externally tangent</td>
<td><img src="image6.png" alt="Figure 6" /></td>
</tr>
<tr>
<td>7. Internally tangent at one point</td>
<td><img src="image7.png" alt="Figure 7" /></td>
</tr>
<tr>
<td>8. Internally tangent at two points</td>
<td><img src="image8.png" alt="Figure 8" /></td>
</tr>
<tr>
<td>9. Transversal at one point and tangent in other</td>
<td><img src="image9.png" alt="Figure 9" /></td>
</tr>
<tr>
<td>10. Coincident</td>
<td><img src="image10.png" alt="Figure 10" /></td>
</tr>
</tbody>
</table>
The coefficients in the above equations may be easily found by comparing them with Equations (2.54) and (2.55), (2.56) (Hughes and Chraibi, 2012). The above polynomials may also be represented in a general matrix form as

\[
\begin{bmatrix}
  x^i & y^i & 1
\end{bmatrix}
\begin{bmatrix}
  a & b & c \\
  b & c & e \\
  d & e & f
\end{bmatrix}
\begin{bmatrix}
  x^i \\
  y^i \\
  1
\end{bmatrix} = 0
\]  

(2.59)

or more conveniently,

\[X^T C_i X = 0 \] (2.60)

\[X^T C_j X = 0 \] (2.61)

The two ellipses will form a pencil of curves given by

\[X^T (\lambda C_i + C_j) X = 0 \] (2.62)

The characteristic equation of the above pencil will be a third degree polynomial,

\[f(\lambda) = det(\lambda C_i + C_j) = \lambda^3 + \lambda^2 l + \lambda m + n \] (2.63)

53
where \( \lambda \) is any parameter and \( l, m \) and \( n \) are constants. The relative position of two ellipses can be determined by analyzing the roots of the characteristic polynomial. Etayo et al. (2006) proved that the nature of the roots is governed only by the signs of the below polynomials

\[
\begin{align*}
    s_1 &= l \\
    s_2 &= -3m + l^2 \\
    s_3 &= 3ln + ml^2 - 4m^2 \\
    s_4 &= -27n^2 + 18lnm + l^2m^2 - 4l^3n - 4m^3
\end{align*}
\]

Etayo et al. (2006) have correlated all the relative positions of polynomials using the above polynomial. For the purpose of this work, however, only relative position 2 in Table 2.2 is of interest, where the ellipses have two real intersection points. For this condition to occur the characteristic Equation (2.63) must have one real negative root and two complex conjugate roots. This is achieved by only the following set of conditions

\[
\begin{align*}
    s_1 > 0; s_4 < 0 \\
    s_1 \leq 0; s_3 \leq 0; s_4 < 0 \\
    s_1 > 0; s_2 = 0; s_3 = 0; s_4 = 0
\end{align*}
\]

Note that relative position 6 in Table 2.2 i.e. two ellipses being externally tangent, is not considered a collision event because there is no overlap and hence will not generate any collision forces according to soft sphere dynamics. Starting from an initial position
where all particles are separated, all other collision scenarios in Table 2.2 are not viable and hence need not be considered in the current work.

### 2.3.1.2 Particle-Wall Overlap Detection

The technique for detecting particle wall overlap is similar to particle particle collision detection. The wall boundary for channel flow can be represented by an equation for a straight line

\[ y = mx + c \]  
 \[ (2.71) \]

where \( m \) is the slope of the line and \( c \) is the y-intercept. The above equation needs to be converted in the coordinate system fixed to the center of the particle. Thus, using Equation (2.2) we get

\[ x = M^i x^i + x_P^i \]  
 \[ (2.72) \]

Using Equations (2.71) and (2.72) we have the equation of the wall-boundary in the body coordinate system

\[ y^i = m^i x^i + c_i \]  
 \[ (2.73) \]

Equations (2.54) and (2.73) may be solved simultaneously to yield a quadratic in \( x^i \)

\[ x^i (A_i^2 m^2 + B_i^2) + x^i 2m_i c_i A_i^2 + A_i^2 c_i^2 - A_i^2 B_i^2 = 0 \]  
 \[ (2.74) \]

The above equation will have real roots only if its discriminant is greater than 0. The case where roots are real and equal is not considered as a collision event since there will
be no overlap. Thus for collision to occur,

\[ D = 2m_i c_i A_i^2 - 4 (A_i^2 m_i^2 + B_i^2) (A_i^2 c_i^2 - A_i^2 B_i^2) > 0 \] (2.75)

Note that the above two sections focus on collision detection among two planar ellipses and ellipse-wall, where wall is represented by a straight line. Extending the analysis into three dimensions may be implemented in a similar way with the difference that the characteristic equations will be one order higher (see Chapter 4).

### 2.3.2 Rough Wall and Rough Particle Modelling

Particles suspended in internal flows with size comparable to the amplitude of wall roughness inevitably suffer from irregular bouncing of the walls. Kussin and Sommerfeld (2002) and Sommerfeld and Kussin (2004) performed extensive experiments with glass beads as particles suspended in a channel flow with untreated rough wall boundaries. They reported that the roughness caused significant re-suspension of the particles, thus changing the concentration profile across the height of the channel. In addition the roughness also caused a reduction in the mean horizontal velocity of the particles and also changed the turbulence structures of Eulerian field by increasing turbulence dissipation.

With the objective of modelling wall roughness in a numerical simulation, Sommerfeld (1992) suggested stochastic modelling of the roughness angle based on a Gaussian distribution function. This model was an improvement over Sommerfeld (1990) wherein the roughness angle was modelled from a uniform distribution function, which was later found out to be inconsistent with the measured values of coefficient of restitution through experimentation. Earlier attempts to model rough wall effects include MATSUMOTO and SAITO (1970), who modelled a rough wall as a sinusoidal wave with a randomly sampled
phase, and Tsuji et al. (1985), Tsuji et al. (1987), who considered virtual rough wall with a random inclination angle. Sommerfeld and Huber (1999) corrected their earlier model by accounting for what they called the ”shadow effect”. This phenomenon is based on the observation that for a particle hitting the wall with an angle of inclination $\alpha_1$ (see Figure 2.3) with respect to the horizontal wall, a negative roughness angle, $\gamma$, of magnitude greater than $\alpha_1$ will lead to an unrealistic collision event. Hence the probability distribution function, i.e. a Gaussian distribution, is modified to exclude all negative angles greater than the incidence angle. Figure 2.4 shows how the normal distribution function is modified for different incident angles, as reported by Sommerfeld and Huber (1999).

![Figure 2.3: Illustration of Shadow effect due to wall roughness (Sommerfeld and Huber, 1999)](image)

Mallouppas and van Wachem (2013) adopted the above model suitably and applied it to the soft-particle approach. Since actual overlap exists in this method a virtual wall needs to be generated at some distance from the actual wall to avoid particle volume
Figure 2.4: Modification of normal distribution function for different incident angles
(Sommerfeld and Huber, 1999)
from appearing outside the domain during collision. The virtual wall is introduced when the shortest distance between the particle and the actual wall was equal to the wall roughness amplitude, which was taken to be 10% of the particle diameter. A random wall roughness angle was then sampled, considering the shadow effect, as discussed earlier. The standard deviation for the distribution is given by actual roughness of wall. These values are enumerated by Sommerfeld and Huber (1999). The virtual wall was then rotated through the chosen roughness angle and the collision dynamics were determined based on this wall. The wall roughness angle remained the same till all time integration steps of collision. In case the smallest distance between the particle and the original wall dropped to half of the distance between the first virtual wall and the original wall, a new virtual wall was generated and a new roughness angle was sampled. New walls were successively added at half the distance till the particle was moving away from the wall (see Figure 2.5). Sommerfeld and Huber (1999) reported that their simulations required three virtual walls for all collision events. Thus, in this work also three virtual walls are considered.

Figure 2.5: Virtual wall generation as proposed by Sommerfeld and Huber (1999)
Collision among rough particles is less popularly discussed in the literature as compared to collision of particle with a rough wall. For example, Allen and Tildesley (1989) briefly discussed the mechanics of collision of rough spheres in the light of molecular dynamics. Physically, as illustrated in Figure 2.6, the difference between a rough and smooth sphere is that the tangent to the surface of a rough sphere will not be perpendicular to the radial vector of the sphere at the collision point. Thus the tangent or collision plane may be modelled using a probability distribution function in a similar way as was done for rough walls. Following the rough wall analysis, normal distribution is used for sampling the collision plane for rough particle-particle collision. Experimental investigation will be required to figure out the exact distribution function, which depends on the type of particles considered for analysis. Also, owing to the convex nature of the surfaces of both colliding particles it is a reasonable conjecture that the "shadow effect" will not apply to particle-particle collision.

Figure 2.6: Plane along which the collision mechanics are modelled for smooth and rough spheres
2.3.3 Collision Forces and Torques

2.3.3.1 Spherical Particles

In the soft sphere approach the collision forces are modeled based on the actual material properties of the particles. Mindlin and Deresiewica (2014) were the first to give a relation for the contact forces in soft sphere approach by modelling the collision process as an equivalent spring compression system. Cundall and Strack (1979) improved on this model by physically describing the collision as a non-linear spring-dashpot-slider system (see Figure 2.7).

![Figure 2.7](image)

Figure 2.7: (a) Spring mass damper system for modelling normal contact force. (b) Spring mass damper system for modelling tangential contact force.

Thus the contact forces according to Hertzian contact theory are given by

\[
\begin{align*}
\vec{F}_n & = -k_n \delta^{3/2} \vec{n} - \eta_n \vec{q}_n \\
\vec{F}_t & = \begin{cases} 
-k_t \vec{\delta}_t - \eta_t \vec{q}_t & \text{if sticking i.e.} |\vec{F}_t| < \mu |\vec{F}_n| \\
-\mu |\vec{F}_n| \frac{\vec{q}_t}{|\vec{q}_t|} & \text{if sliding i.e.} |\vec{F}_t| \geq \mu |\vec{F}_n| 
\end{cases}
\end{align*}
\]
where $\vec{q}_n$ and $\vec{q}_t$ are the relative velocities of particle $j$ with respect to particle $i$ in the normal and tangential planes respectively. $k_n$ and $\eta_n$ are the spring stiffness and damping coefficient respectively in the normal direction and $k_t$ and $\eta_t$ are the respective coefficients in the tangential direction. The material properties of the particles are related to the spring-dashpot system as

\begin{align}
  k_n &= \frac{4}{3} \left( \frac{1 - \sigma_{p1}^2}{E_{p1}} + \frac{1 - \sigma_{p2}^2}{E_{p2}} \right)^{-1} \left( \frac{r_{p1} + r_{p2}}{r_{p1}r_{p2}} \right)^{-\frac{1}{2}} \tag{2.78} \\
  k_t &= 8 \left( \frac{2 - \sigma_{p1}}{G_{p1}} + \frac{2 - \sigma_{p2}}{G_{p2}} \right)^{-1} \left( \frac{r_{p1} + r_{p2}}{r_{p1}r_{p2}} \right)^{-\frac{1}{2}} \delta_n \tag{2.79}
\end{align}

where $E$ and $G$ are the Young’s Modulus and Shear Modulus of the respective particles while $\delta_n$ and $\delta_t$ are the deformations in the normal and tangential direction.

$$G = \frac{E}{2(1 + \sigma)}$$

Normal deformation is clearly depicted in Figure 2.7(a). Tangential deformation on the other hand is calculated by integrating the tangential velocity over collision time steps,

$$\delta_t = \int q_t dt \tag{2.80}$$

Tsuji et al. (1992) improved on the relations given by Cundall and Strack (1979) for connecting the damping coefficient of the equivalent dashpot with the material properties of particles. Their model is based on the analytical analysis relating the damping coefficient to the coefficient of restitution. Thus,
\[ \eta_n = \alpha \sqrt{M n} \delta^n \]  
\[ \eta_t = \alpha \sqrt{M k} |\delta_n^{1/2}| \]  

where \( M \) is the equivalent mass of the pair of particles given by,

\[ M = \frac{m_{p1} m_{p2}}{m_{p1} + m_{p2}} \]

while \( \alpha \) is the coefficient that depends solely on the coefficient of restitution, as introduced by Tsuji et al. (1992) (see Figure 2.8).

Figure 2.8: Coefficient \( \alpha \) as a function of restitution, as calculated by Tsuji et al. (1992)

Note that for collision with walls \( r_{p2} = r_w \to \infty \), thus, \( \frac{r_{p1} + r_w}{r_{p1} r_{w}} = \frac{1}{r_{p1}} \). Also \( m_w \to \infty \), hence \( M = m_{p1} \).
2.3.3.2 Ellipsoidal Particle Interaction (EPI) Model

This section introduces the proposed model which is intended to replace the multi-sphere approach for ellipsoid interaction. After the collision between ellipsoid pairs or ellipsoid and wall has been established as explained in Section 2.3.1.1, the first step in finding collision forces is to determine the collision point of two overlapping ellipses.

1. Contact Point Calculation:

Džiugys and Peters (2001a) enumerated three different algorithms to achieve this. The intersection algorithm was developed by Rothenburg and Bathurst (1991) and Ting (1990). The contact point according to the intersection algorithm was the mid point of intersection points of the overlapping ellipses (Figure 2.9(left)). This method has been reported to have accuracy and stability issues in cases where the overlap area is too small or when axes of the ellipses are aligned to each other. Also, this algorithm cannot be transported to ellipsoids. Ting (1992) developed the geometric potential algorithm. The contact point is given by the midpoint of "touch" points \( T_i \) and \( T_j \) which are the deepest points, or the points with the highest potential, on an ellipse inside the other ellipse. This method can be conveniently applied to ellipsoids and also offers more accuracy and stability (Figure 2.9 (right)). Common normal algorithm, introduced by Lin and Ng (1995), is described by Figure 2.10 and is based on solving for the common normal of two ellipses. It was however discovered that the geometric potential algorithm is the most stable of the three and is therefore used the most in the literature.

In this work the method used for finding the collision point is similar in concept to the geometric potential algorithm i.e. finding the point on ellipse with the highest potential in the other ellipse, but is calculated more conveniently by reducing the
Figure 2.9: Collision point according to the (left) intersection and (right) geometric potential algorithm. The geometrical potential algorithm is also applicable to 3-D ellipsoids.

Figure 2.10: Collision point according to the common normal algorithm. Extendable to 3-D ellipsoids.
problem to finding the local maxima or minima of the curve. This is illustrated in Figure 2.11. The equation of both the ellipses is carefully transformed from the respective body fitted coordinate systems into a coordinate system \((x', y')\) with \(x'\) axis aligned with the line joining the intersection points. The maxima/minima on the ellipse arc would then give the "deepest" points on both ellipses. In contrast to the algorithms listed above, the collision point, however, is not chosen to be the mid point of the line joining \(T_i\) and \(T_j\). Instead, to keep the collision physics more akin to the case of colliding spheres and perhaps more realistic, the line joining the intersection points \(A, B\) is taken to be the extent to which colliding ellipses deform during collision. Thus, a more appropriate choice for the collision point will be the point of intersection of line joining \(A, B\) and the line joining the "deepest" points \(T_i\) and \(T_j\), as illustrated in Figure 2.12.

2. Calculation of interacting 'psuedo-spheres':

After the collision point has been calculated the next problem is to determine the equivalent deformation \(\delta_n\) which is to be input to Equation (2.76) for the calculation of normal force and also equivalent radii, \(r_{p_i}\) and \(r_{p_j}\) for calculating the stiffness of the equivalent spring dashpot system. Note that the soft-particle theory by Cundall and Strack (1979) applies only to spherical particle. Again in the Multi-sphere approach as the ellipses are formed by fusing several spheres together, the required radius for calculating the collision forces is simply the radius of the sphere that is involved in interaction out of all the spheres that form the ellipsoidal particle. Hence the deformation can be conveniently calculated by obtaining the amount of overlap for this interacting sphere. In this study however, instead of having a fixed number of spheres to form the ellipsoidal particle, a fictitious ‘psuedo-sphere’ is
Figure 2.11: The maxima and minima points on colliding ellipses after coordinate transformation

Figure 2.12: (a) Deformation in spheres during collision (b) Deformation in ellipses during collision and the selection of collision point. The dashed arcs represent the original shapes
generated as explained in the following paragraph.

It is convenient to solve for the collision forces and torques in a coordinate system aligned with the collision plane. This system will be referred to as norm-tan \((\vec{n} - \vec{t})\) frame, shown in Figure 2.13. The line joining the intersection points serves as the collision plane (for smooth ellipsoids). Thus the tangent axis of the norm-tan frame is aligned with this line and the normal axis is oriented such that it always points outside of the particle \(i\). Now, a 'pseudo-sphere' is introduced which has its center located on the major axis of the ellipse. The location of its center is determined by finding the intersection of major axis with the line perpendicular to selected collision plane passing through the chosen collision point. This sphere serves as the "interacting" sphere equivalent to what was modeled using the multi-sphere approach. However, the stark difference between the two approaches is that since the interacting psuedo-sphere is determined analytically in the present model, the ellipsoid can be assumed to be composed of infinite such spheres which would eliminate any accuracy issues encountered by the limited number of spheres used to model using the multi-sphere approach. Also this model would eliminate the need for tracking each of these spheres, thus eliminating all the extra memory load requirements that exist in the multi-sphere method.

3. Calculation of radius of psuedo-sphere and \(\delta_n\):

The radius of the interacting circle \(r_p\) and the deformation \(\delta_n\) still need to be determined. The equation of the ellipse and the line joining center of the interacting circle and the collision point are solved simultaneously to generate a point on the ellipse. The line segment joining this point and the center of interacting circle serve as the equivalent radius \(r_p\) for calculating the stiffness from Equation 2.78 and the
Figure 2.13: Calculation of deformation $\delta_{ni}$ and the equivalent radius $r_{pi}$ for calculation of collision forces. A similar process is repeated in the body coordinate system of particle $j$ to find $\delta_{nj}$ and $r_{pj}$. 

Calculated interacting "pseudo-spheres" which are used to determine collision forces 

Line perpendicular to collision plane and passing through collision point C 

Calculated center of interacting "pseudo-sphere" on the major axis of ellipsoid i
line segment joining the collision point and the above calculated point gives the
required deformation $\delta_n$ for calculating collision force in Equation (2.76). The same
process is carried out for calculating equivalent radii, $r_{p_i}$ and $r_{p_j}$, and deformations
$\delta_{n_i}$ and $\delta_{n_j}$ for both particles in their respective body fixed coordinate systems. The
deformation used in Equation (2.76) is the sum of the two deformations.

$$\delta_n = \delta_{n_i} + \delta_{n_j}$$ (2.83)

It is not hard to see that if the particles are spheres, the interacting psuedo-spheres
will coincide with the actual particles. Thus, the above model is equally valid for
modelling collision among spherical particles, although it may prove to be more
computationally expensive as compared to the traditional methods for spheres.

2.3.3.3 Consideration of Rough Ellipses

Section 2.3.2 introduced the physics of how rough spheres are modelled. The same prin-
ciple can be extended to modelling collision of rough ellipses. The collision plane for
smooth ellipses, line joining points $A$ and $B$, is rotated by a randomly sampled angle
about the collision point $C$. Thus the sample space for collision plane will be as shown
in Figure 2.14. The interacting psuedo-sphere is then chosen by the same method as
described in previous subsection. It can be seen from Figure 2.14 that the location and
radius of psuedo-sphere change based on the selection of collision plane leading to differ-
ent collision forces. Similar mechanics are applied for the collision of rough ellipse-wall
collision.
2.4 Code Structure

A C++ based program is developed to model the transport of particles in turbulent channel flow. The flow field was solved using OpenFOAM’s LES solver. The grid for the simulations used in this work was Cartesian type with hexagonal block cells. Irrespective of the grid type however, OpenFOAM stores the information as unstructured mesh, by also storing the cell connectivity information along-with. A program was written to convert the solved flow field to structured format in order to conveniently program the particle transport code. Flow output files were written at fixed time intervals.

Figure 2.15 shows the overall flow of the particle transport code. The initial particle
file is generated by a separate program which utilizes a random number generator to uniformly distribute particles in the domain. In order to fully resolve the collision events but keep the overall computation inexpensive, the particle code is divided into two time stepping mechanisms namely, outer and inner time loop. The division of the two time loops is based on the Near-Neighbor lists (NNL). The NNL also called the Verlet list (Verlet, 1967) were introduced for molecular dynamics simulations. If collision between all combinations of particle pairs is evaluated in the domain, the computation becomes very expensive. To circumvent this a cut-off distance is specified beyond which particle collision is improbable. In the present code the exact formulation used by Kloss et al. (2012) is used to generate NNL.

\[ |\vec{x}_i - \vec{x}_j| < r_i + r_j + s \]  

(2.84)

where \( \vec{x} \) denotes the particle positions and \( r \) their radii. \( s \) is the Verlet parameter. It determines the number of steps, \( N_{\text{verlet}} \), for which the NNL is valid,

\[ N_{\text{verlet}} = \frac{s}{2v_{\text{max}} \delta t} \]  

(2.85)

where \( v_{\text{max}} \) is the maximum expected magnitude of velocity of the particle and \( \delta t \) is the time step size. Note that for ellipsoid particle simulations \( r \) in Equation 2.84 is taken equal to the major axis of the ellipsoid.

Separate NNL’s are created for particle-particle and particle-wall collision for easy identification. Particles that appear in any of the NNL are placed in the inner time loop. Collision detection is performed only on the particles in the inner loop. The time step size
for the inner loop is smaller than the outer loop in order to resolve the collision events properly. For a detailed discussion on time step selection refer Chapter-3, Section-3.3.2. The particles in the inner loop are integrated in time until it catches up with the current outer loop time. For the particles in the outer loop only hydrodynamic drag and lift force and torque need to be evaluated and then the particle positions and velocities are integrated.

2.5 Chapter Summary

Particle transport modelling in flow suspensions are discussed in this chapter. Relations for drag, lift and torque models used for both spherical and non-spherical particles are given. All aspects related to particle impact dynamics, including collision detection and collision force modeling are also detailed for both spherical and non-spherical particles. A novel collision model based on the analytic equations of ellipsoid and the multi-sphere approach is introduced. The model is called the ellipsoid interaction (EPI) model. In the EPI model the collision point is calculated via the geometric potential algorithm (Ting, 1992) which is more accurate than the MS method. The collision forces on the other hand are calculated using a novel concept of interacting ‘psuedo-spheres’ which allows for convenient force modelling akin to the MS method. Modelling approach for rough ellipsoids is also discussed based on the rough wall collision model by Sommerfeld (1992)
Figure 2.15: Flowchart of the EPI code
Chapter 3

Results and Discussion

3.1 Introduction

In the previous chapter the transport of non-spherical particles in turbulent flows is discussed. The particles are treated as discrete elements and their dynamics are governed by Newton’ Laws of Motion. This approach is referred to as Lagrangian phase modeling. The primary objective of this work, however, is to introduce a novel collision model for non-spherical particles, particularly ellipsoids of different aspect ratios, which would be an improvement over the existing Multi-Sphere (MS) approach. The MS method is currently being used as a standard for implementing collision dynamics for non-spherical particles, particularly for modelling particle-laden flows. The theoretical details of this new model, named the Ellipsoidal Particle Interaction (EPI) approach are given in Chapter-2.

A major hurdle in the study of non-spherical particles is the non-availability of sufficient experimental data. Majority experimental investigations available in the open literature exist only for simple flow scenarios i.e., horizontal/vertical channels or hor-
izontal/vertical pipes. A number of phenomena need to be accounted for with the introduction of particles in these flow regimes (Sommerfeld, 2003) e.g., particle transport, slip-shear lift force on particles in turbulent boundary layers (Mei, 1992; Wang et al., 1997), Magnus effect acting on the particles due to high angular velocities after wall collisions, wall roughness effects, inter-particle collisions and turbulence modification due to high concentration of particles. Sommerfeld (2003) enumerated the various experimental studies done in the field, each study focusing on one or two particular phenomena mentioned above. It must, however, be pointed out that all these experimental studies exist for spherical particles only. From an application standpoint, especially in toxic or therapeutic particle transport and deposition, non-spherical particles are far more important. The experimental data for spherical particles can however serve as a benchmark for computational analysis of non-spherical particles. Computational study of non-spherical particle transport in turbulent flow fields is a subject of very recent research. Table 3.1 lists some of the computational studies carried out for simulating non-spherical particles in turbulent gas-particle flows with horizontal channel configurations.
<table>
<thead>
<tr>
<th>Study</th>
<th>Geometry Dimension</th>
<th>(Re_T)</th>
<th>Particle Aspect Ratio</th>
<th>Inter-Particle Collision</th>
<th>Flow Solver</th>
<th>Fluid-particle coupling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Challabotla et al. (2015)</td>
<td>(1.5h \times 0.75 \times h)</td>
<td>180</td>
<td>0.01, 0.1, 0.33, 0.5 and 1</td>
<td>No</td>
<td>DNS</td>
<td>One-way coupling</td>
</tr>
<tr>
<td>Zhao et al. (2014)</td>
<td>Different dimensions considered</td>
<td>150, 180 and 300</td>
<td>1, 3, 10 and 50</td>
<td>No</td>
<td>DNS</td>
<td>One-way coupling</td>
</tr>
<tr>
<td>Zhao and van Wachem (2013a)</td>
<td>(4\pi h \times 2h \times 2\pi h) (h = 0.02m)</td>
<td>150</td>
<td>1, 3</td>
<td>Yes</td>
<td>DNS</td>
<td>Fully Coupled</td>
</tr>
<tr>
<td>Marchioli et al. (2008)</td>
<td>(4\pi h \times 2h \times 2\pi h)</td>
<td>150</td>
<td>1</td>
<td>No</td>
<td>DNS</td>
<td>One-way coupling</td>
</tr>
<tr>
<td>Marchioli et al. (2010)</td>
<td>(4\pi h \times 2h \times 2\pi h)</td>
<td>150</td>
<td>Range of fibers considered</td>
<td>No</td>
<td>DNS</td>
<td>One-way coupling</td>
</tr>
<tr>
<td>Mortensen et al. (2008)</td>
<td>(1.5h \times 0.75 \times h)</td>
<td>360</td>
<td>Range of aspect ratios considered</td>
<td>No</td>
<td>DNS</td>
<td>One-way coupling</td>
</tr>
<tr>
<td>van Wachem et al. (2015)</td>
<td>(175mm \times 35mm \times 35mm)</td>
<td>600</td>
<td>1, 2.5, 1.25, 5</td>
<td>Yes</td>
<td>LES</td>
<td>Fully Coupled</td>
</tr>
</tbody>
</table>

Table 3.1: Existing computational studies on particle-laden channel flow
An obvious choice for benchmarking the EPI model were the simulations conducted by van Wachem et al. (2015) since they considered inter-particle collisions and different particles including spheres, ellipsoids, fibers and discs. They referenced experimental results put forth by Kussin and Sommerfeld (2002) for spherical particles to support their computational data for non-spherical particles.

Figure 3.1: Description of the test facility used by Kussin and Sommerfeld (2002)

Figure 3.1 shows the sketch of the experimental setup used by Kussin and Sommerfeld (2002). The experiments were performed for particle mass loading of 1 to 2 (particle mass loading is the ratio of particle mass to gas mass in the flow domain). The objective of the experiments was to investigate the effects of wall-roughness of varying degrees (different wall materials were used) on the particle concentration profile measured across the width of the flow channel. The horizontal channel had the dimensions $6m \times 35mm \times 35mm$ and the particles considered were spherical glass beads and had diameters in the range $60\mu m-1mm$. 
Mallouppas and van Wachem (2013) performed two-way coupled LES simulations with particle laden flows to confirm the experimental work of Kussin and Sommerfeld (2002) computationally. The particles considered by them were only spherical. They presented simulation results for both smooth and rough channel walls. Furthermore, the effects of considering one-way coupled and two-way coupled modeling approaches on particle concentration and mean particle velocities were also compared. For modeling inter-particle and particle wall collisions they considered both hard-sphere and soft-sphere techniques and presented a comparison between the two. It was concluded that the differences in mean particle velocity profiles between the one-way and two-way coupled simulations were insignificant. Thus, for the particle mass loading under consideration, the simulations may be considered to be one-way coupled without sacrificing significantly on accuracy. The differences in concentration profile for hard-sphere and soft-sphere approach were, again, insignificant. However, the hard-sphere approach is suitable only for fast-moving dilute flows because hard sphere modelling technique is based on empirical data which, inevitably, leads to errors. In contrast, the soft-sphere formulation is based only on the mechanical properties of particles and hence is generally more accurate and more widely applicable in terms of concentration of particles considered (see Chapter-1, Section 1.3.1.2.1).

Using the simulations of Mallouppas and van Wachem (2013) as basis, van Wachem et al. (2015) extended their analysis to non-spherical particles by considering ellipsoids of five different aspect ratios suspended in the same turbulent regime. Table 3.1 enumerates the particles that were considered. The sphericity of a particle gives a measure of how rounded or close to spherical the particle is. It is defined as,

\[ \text{Sphericity} = \frac{A_{\text{sphere}}}{A_{\text{particle}}} \]  

(3.1)
where $A_{sphere}$ is the surface area of the sphere with the same volume as the particle and $A_{particle}$ is the surface area of the particle. All simulations performed by van Wachem et al. (2015) were two-way coupled and both smooth and rough boundary walls were accounted for. Their results showed that the concentration and velocity profiles of non-spherical particles were significantly different as compared to spherical particles, emphasizing the importance of non-spherical modelling.

The objective of the present work is to develop a novel collision model for axisymmetric non-spherical particles and apply it to particle-laden flow simulations. An in-house C++ based was developed for this purpose from scratch. In order to minimize programming efforts the methodology was applied only to in-plane (2-D) collisions. Simulations by van Wachem et al. (2015) and Malloupas and van Wachem (2013) feature flow in an infinite channel. Thus, at quasi-steady state the flow can be considered to be essentially 2-D. The results reported by these studies can therefore be reliably considered for 2-D flow simulations as well. Following successful validation the EPI model can be adopted with confidence in a 3-D code. Details on collision modelling for 3-D ellipsoids are covered in Chapter-4, Section 4.2.1. The following sections describe the methodology for 2-D simulations.
### Table 3.2: Different particles considered for analysis by van Wachem et al. (2015)

<table>
<thead>
<tr>
<th>Name</th>
<th>Shape</th>
<th>Sphericity</th>
<th>Aspect Ratio</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sphere</td>
<td><img src="image" alt="Sphere" /></td>
<td>1</td>
<td>( \frac{a}{b} = 1 )</td>
<td>200µm</td>
</tr>
<tr>
<td>Ellipsoid 1</td>
<td><img src="image" alt="Ellipsoid" /></td>
<td>0.885</td>
<td>( \frac{a}{b} = \frac{5}{2} )</td>
<td>( a = 368µm ) ( b = 147µm )</td>
</tr>
<tr>
<td>Ellipsoid 2</td>
<td><img src="image" alt="Ellipsoid" /></td>
<td>0.991</td>
<td>( \frac{a}{b} = \frac{5}{4} )</td>
<td>( a = 232µm ) ( b = 68.4µm )</td>
</tr>
<tr>
<td>Disc</td>
<td><img src="image" alt="Disc" /></td>
<td>0.626</td>
<td>( \frac{a}{b} = 5 )</td>
<td>( a = 342µm ) ( b = 68.4µm )</td>
</tr>
<tr>
<td>Fiber</td>
<td><img src="image" alt="Fiber" /></td>
<td>0.639</td>
<td>( \frac{a}{b} = 5 )</td>
<td>( a = 510µm ) ( b = 102µm )</td>
</tr>
</tbody>
</table>

### 3.2 Flow Field Solution

As mentioned earlier, Mallouppas and van Wachem (2013) concluded from their simulations that the difference between results obtained for one-way coupled and two-way coupled simulations were not substantial. Therefore, current simulations are designed to be one-way coupled, circumventing the need to develop an Eulerian flow field solver for turbulent flows in order to couple the momentum source terms of Lagrangian (particle) phase with Eulerian phase. The LES solver offered by OpenFOAM was used in this study to solve for the turbulent flow field. The flow parameters were chosen to be same as described by Mallouppas and van Wachem (2013) and van Wachem et al. (2015). The fluid under consideration was air having a dynamic viscosity \( \mu_f = 18.62 \times 10^{-6} Pa.s \) and
density $\rho_f = 1.15 kg/m^3$. The average air velocity in the channel was $U_{avg} = 19.7 m/s$. Thus the Reynold’s Number defined as:

$$Re = \frac{\rho U_{avg} L}{\mu_f}$$

(3.2)

for the flow was 42,585, where L was the channel height (= 0.035 m).

### 3.2.1 Simulation Setup

For an LES simulation only the large scales of turbulence need to be resolved. The smaller scales are solved for by the so called Sub-Grid Scale (SGS) models (Piomelli, 1997). Thus, the requisite grid size requirements for LES are determined by the length of larger scales. Piomelli (1997) summarized the grid size requirements for LES of channel flows. In order to resolve the laminar-sub layer region of the turbulent flow, LES requires a DNS type resolution near the walls. Thus, the first grid point must be located at $y^+ \sim 1$. Away from the wall, in the turbulent region, larger mesh cells may be used. The grid spacing requirements in the length-wise ($x$) and span-wise ($z$) directions for LES are $\Delta x^+ \sim 50 - 150$ and $\Delta Z^+ \sim 15 - 50$. The dimensionless channel height is defined as:

$$y^+ = \frac{\rho U_* \Delta y}{\mu}$$

(3.3)

where $\Delta y$ is the distance from the wall and $U_*$ is the friction velocity defined using the wall shear stress $\tau_w$ as,

$$U_* = \sqrt{\frac{\tau_w}{\rho}}$$

(3.4)
For turbulent flow simulations a more convenient Reynolds’s number based on the friction velocity is defined, known as the friction Reynolds’s number.

\[ Re_\tau = \frac{\rho U^* L}{\mu} \]  \hspace{1cm} (3.5)

where \( L \) is the reference length, which is commonly taken as the half-channel height for channel flows. For the simulations performed by van Wachem et al. (2015) \( Re_\tau = 600 \). Thus for LES the required maximum grid cell height next to cell, corresponding to \( y^+ = 1 \) is \( \Delta y \approx 0.03mm \). Similarly, limit for the grid spacing in length-wise direction was \( \Delta x = 3mm \) (corresponding to \( x^+ = 100 \)) and in the span-wise direction was \( \Delta z = 1.5mm \) (corresponding to \( z^+ = 50 \)).

The dimensions of the channel were kept similar to van Wachem et al. (2015) i.e., \( 0.175m \times 0.035m \times 0.035m \) (see Figure 3.2). Following the preceding grid size limitations two meshes were generated with different refinement levels. The meshes were generated using OpenFOAM’s \textit{blockMesh} utility. From the center of the channel a refinement ratio of 5 was applied towards the channel walls in the \( y \) direction (see Figure 3.3b). Consequently there were atleast 5 mesh points in the \( y^+ = 10 \) region for both meshes. To summarize, all cell dimensions for both meshes were within the bounds required for LES simulation. Figure 3.3 shows the mesh with 500,000 cells. The details for the two meshes are listed in Table 3.3.
Figure 3.2: Geometry used in the simulations. Periodic boundary conditions (BC) are specified on the left-right and front-back faces.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>No. of elements</th>
<th>Cell length in x ($\Delta x$)</th>
<th>Cell length in y (Near Wall $\Delta y$, Channel Centre $\Delta y$)</th>
<th>Cell length in z ($\Delta z$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>500000</td>
<td>0.7mm</td>
<td>0.012152 mm, 0.972 mm</td>
<td>1.4 mm</td>
</tr>
<tr>
<td>2</td>
<td>820000</td>
<td>0.5833mm</td>
<td>0.0038935 mm, 0.79 mm</td>
<td>1.1667 mm</td>
</tr>
</tbody>
</table>

Table 3.3: Generated Mesh Specifications
3.2.2 Boundary conditions and Flow Initialization

Periodic boundary condition were specified on the front-back face pair and also the left-right face pair. These settings effectively simulate flow between parallel plates with infinite length. In real geometries the initial turbulence structures arise in the channel due to imperfections in the wall. These small natural perturbations grow in scale over time to large scale turbulent structures throughout the channel. However while simulating LES since there is no source for initial near-wall perturbation cycles, they have to be artificially introduced. Theoretically, perturbations arising from round-off machine errors may also evolve into turbulent structures with simulation time. However that requires that the simulation is carried out for a very long time period. A much efficient solution was introduced by De Villiers (2007). The \textit{perturbUChannel} utility in OpenFOAM introduces small artificial perturbations near the channel walls. This utility was used for flow initialization for LES simulations. It was however observed that with the required dynamic
viscosity of $18.62 \times 10^{-6} Pa.s$ the initial perturbations were attenuated with time resulting in a laminar type profile. The simulation was thus run initially with $18.62 \times 10^{-7} Pa.s$. After the turbulent structures were substantial the viscosity was adjusted back to original value.

As mentioned earlier the LES solver does not resolve the small length scales. Specifically, a cut-off filter ($\Delta$) is applied and the length scales smaller than the filter are modelled using so called sub-grid scale (SGS) models. In the simulations for this work the Smagorinsky (1963) model is used for modelling SGS stresses with van Driest damping for eddy viscosity near the walls.

### 3.2.3 Single-phase simulation results

Figure 3.4 shows the stream-wise component of velocity averaged over time for both meshes. The stream-wise velocity was compared with both the results reported by Mallouppas and van Wachem (2013) for one-way coupled LES simulation with Smagorinsky model and with experimental results reported by Kussin and Sommerfeld (2002) for spherical particle laden experiments. The simulations in the present work were run for a total of $0.5$ secs with maximum Courant number of 1. From the figure (3.4) it is evident that both the meshes generate good results in comparison to the velocity profile reported by Mallouppas and van Wachem (2013). This was expected since the grid resolution for both meshes was within the size requirements for LES simulation. Note that the mean velocity for experiment with particle suspension was lower as compared to the single phase simulations/experiment due to the momentum transfer from fluid phase to the particles. Figure 3.5 shows the instantaneous velocity contours of the solved flow field at
Figure 3.4: One-way simulation result comparison: Average stream-wise mean fluid velocity (in m/s)
(a) Contours of instantaneous stream-wise fluid velocity (in m/s) at $t = 0.5secs$ taken at mid-plane ($z = 0.0175m$)

(b) Contours of instantaneous wall-normal fluid velocity (in m/s) at $t = 0.5secs$ taken at mid-plane ($z = 0.0175m$)

Figure 3.5: Contours of instantaneous fluid velocity in the longitudinal cross-section of channel
3.3 Particle-Laden Simulations

3.3.1 Mapping Eulerian Fields to Lagrangian Domain

In the Eulerian-Lagrangian framework of modelling, as used in this work, the particles are considered as point masses suspended in the flow field. The hydrodynamic forces are interpolated to the location of particle centers and then the particle dynamics are solved using the Newton’s second law of motion with forces acting on the mass centers. This approximation, however, is valid only if the particle diameter is smaller than the Kolmogorov scale. For numerical simulations this implies that the particle diameter must be smaller than the grid cell dimensions (Mallouppas and van Wachem, 2013). Obeying this limitation the fluid properties in the Eulerian framework can be converted to Lagrangian frame by simple interpolation from grid locations to particle positions. Mallouppas and van Wachem (2013) and van Wachem et al. (2015) used cubic spline interpolation for this purpose following the study done by Yeung and Pope (1988). In this work, however, a linear interpolation scheme was used for the purpose of simplicity. Specifically, since the mesh involved was 2-D consisting of rectangular elements, bilinear interpolation was used. Franklin and Lee (2010) reported bilinear interpolation to provide high quality interpolation for structured grids. The disadvantage of linear interpolation scheme is that it ignores non-linear variations that are smaller than the grid size. It provides second-order accuracy and is $C^0$ continuous because of linear polynomial approximation. This implies that the fluid properties at the cell interfaces will be continuous, which is a necessary requirement of an interpolating function (Yeung and Pope, 1988), but the gradients of the flow variables may not necessarily be continuous. Cubic spline interpolation on the other hand provides $C^2$ continuity being of higher order. Some applications of linear
interpolation are Deardorff and Peskin (1970) and Riley and Patterson Jr (1974). The area-weighted averaging interpolation technique adopted by Van Wachem et al. (2001) provides the same numerical accuracy as the linear interpolation scheme. Figure 3.6 gives a geometric description of the bilinear interpolation scheme.

Figure 3.6: Depiction of bilinear interpolation of Eulerian quantities from cell centres to particle position $D$

Through DNS studies conducted by Bagchi and Balachandar (2003) it was reported that the hydrodynamic forces acting on a particle suspended in a flow field are accurately predicted if the particle diameter is in the range $1.5\eta_k < d_p < 10\eta_k$, where $\eta_k$ is the Kolomogorov length scale. Vreman et al. (2009) later conducted four way coupled LES
simulations to confirm that the drag forces are predicted with sufficient accuracy if $d_p \leq 4\eta_k$. Based on experimental data Malloupass and van Wachem (2013) reported that the Kolmogorov length scale for the considered channel flow case was $\eta_k,\text{center} = 9.35 \times 10^{-5} m$ at the centre of the channel and $\eta_k,\text{wall} = 2.911 \times 10^{-5} m$ near the channel walls. As an aside, note that considering these length scales and from Table 3.3 it can be confirmed that all the structures near the channel walls are resolved and thus the resolution near the walls is DNS type. The maximum equivalent diameter of the particles considered in this study (see Table 3.2) was 200$\mu m$. Therefore the drag force is expected to be accurate on the particles throughout the channel. Malloupass and van Wachem (2013) utilized a so called ‘particle mesh’ to map the Eulerian quantities from the actual cartesian mesh. The particle mesh was isotropic with equally spaced cells and overlapped the cartesian mesh. The dimensions of the particle mesh cells were such that cells were larger than the particle diameter, consistent with the requirement of point-particle approach. Generating an isotropic particle mesh is straightforward for simple geometries such as in the present case. However for complicated geometries it is not feasible and requires extra memory. In this work the Eulerian quantities are directly mapped from the Cartesian mesh points to particle position. The mesh cell dimensions in the stream-wise direction and in the center of the channel are sufficiently large as compared to the particle diameter. Only near the channel walls the cell dimensions become smaller than particle dimensions. At these locations the mesh cells are combined to form a larger cell such that it encompasses the particle completely. This is done at run-time rather than storing particle mesh for the the entire flow domain separately. Figure 3.7 shows a geometric description of how the near wall cell data is mapped onto a representative bigger cell.
3.3.2 Simulation Setup

Due to the application of periodic boundary conditions the simulated particulate flow in the channel was essentially 2-D. The transverse velocity component was negligible as compared to the stream-wise or wall-normal velocity for both the particles and the fluid. The parameters of interest studied by Mallouppas and van Wachem (2013) and van Wachem et al. (2015) e.g., the concentration and velocity profile of the particles as a function of channel height, mean and fluctuating velocity of the fluid as a function of channel height etc., were not affected by the transverse movement of the particles. This fact has been used to advantage in the present work to reduce the computational load and make the collision modelling simpler. Interaction between the particles and the particle and walls were restricted to planar collisions by constraining the rotation of the particles in a single plane ($x-y$ plane in Figure 3.2). Any transverse velocity resulting from a collision event was neglected. Owing to the periodic boundary conditions the distribution of the particles in the transverse direction was assumed to be uniform.
3.3.2.1 Particle properties and Time-step consideration

The particles considered were glass beads with a density of ρ_p = 2500 kg/m^3. As mentioned earlier, in the soft particle approach the collision forces are modelled using the mechanical properties of the particles. The problem of collision resolution corresponds to the problem of spring mass damper system in the soft particle theory Tsuji et al. (1992). The equation of motion of such a system is given by

\[ m x'' + c x' + K x = 0 \] (3.6)

where \( m \) is the mass of the particle; \( K \) is the stiffness of equivalent spring and \( c \) is the damping coefficient of the damper. For the above system the undamped natural frequency is given by,

\[ \omega_o = \sqrt{\frac{K}{m}} \] (3.7)

and the damping ratio \( \zeta \) is given by,

\[ \zeta = \frac{c}{2 \sqrt{km}} \] (3.8)

The natural frequency of oscillation of the damped system is,

\[ \omega = \omega_o \sqrt{1 - 2 \zeta^2} \] (3.9)

Thus the overlap time of a binary collision between two particles will be,

\[ t_{overlap} = \frac{\pi}{\omega} \] (3.10)
Schwartz et al. (2012) suggested that about 50 time-steps are required in order to accurately resolve the collision event. Thus the maximum allowed value of time step was given by,

$$\Delta t_{coll} = \frac{t_{overlap}}{50}$$  \hspace{1cm} (3.11)

For the glass particles considered in this work the realistic material properties are (Schwartz et al., 2012) Young’s Modulus $E = 68.9 \, GPa$, Poisson’s Ratio $\nu = 0.24$. Based on these value and from equations 2.78, 2.79 and 3.11, the typical order for a collision event was found to be $t_{overlap} \sim 10^{-8}$. Thus the required simulation time step would have been $\Delta t_{coll} \sim 10^{-9}$. Obviously such small time step selection is impractical for computations. For this reason the particle properties considered for DEM simulations often are scaled down. Several studied have been performed to find equivalent properties for glass particles for DEM simulations (see Alizadeh et al. (2014) for a list of references). In this work the properties suggested by Stewart et al. (2001) have been used, viz., Young’s Modulus $E = 2.16 \, MPa$, Poisson’s Ratio $\nu = 0.24$. For these values the overlap time was found to be $t_{overlap} \sim 10^{-6}$.
Table 3.4: Selected particle properties and time-step values for particle laden simulations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Young’s Modulus ($E$)</td>
<td>2.16 MPa</td>
</tr>
<tr>
<td>Poisson’s Ratio ($\nu$)</td>
<td>0.24</td>
</tr>
<tr>
<td>Coefficient of Restitution ($e$)</td>
<td>0.9</td>
</tr>
<tr>
<td>Coefficient of friction ($\mu_f$)</td>
<td>0.1</td>
</tr>
<tr>
<td>$\Delta t_{coll}$ (Innerloop)</td>
<td>$10^{-8}$ secs</td>
</tr>
<tr>
<td>$\Delta t_{sim}$ (Outerloop)</td>
<td>$5 \times 10^{-6}$ secs</td>
</tr>
<tr>
<td>Density ($\rho_p$)</td>
<td>2500 kg/m$^3$</td>
</tr>
</tbody>
</table>

Table 3.4 enumerates the parameters selected for particle simulations. Note that there are two different time step values listed in the table. The time-stepping algorithm is designed such that the particles that are in the near-neighbour list (NNL) evolve at a smaller time-step as compared to particles that are not and hence are not likely to collide. This selective time-stepping mechanism allows for complete resolution of the collision events without the algorithm being too computationally expensive as a whole.

In all the simulations the particles were initialized in the flow field with a uniform distribution. The initial velocity of the particles was assigned such that they had a small relative velocity with respect to the surrounding fluid. The initial particle orientation for the non-spherical particles was randomly assigned. All simulations were run on the NCSU HPC facility on 8 cores.
3.3.3 Smooth Wall Simulations: Spherical Particles

For the purpose of benchmarking the EPI code was run for spherical particles and compared with the computational results of Mallouppas and van Wachem (2013). Several computations were performed using different number of particles in the domain to determine the representative number of particles in 2-D that give the same concentration as in 3-D simulations. Two major mechanisms influenced the y-momentum and hence the position of the particles. Particles settled under the effect of gravity and were kept in suspension by means of inter-particle collision. Figure 3.8 compares the position of particles at t=0.5 secs for simulations with and without inter-particle collision, thus illustrating the importance of modelling collision for particle suspension. If the particle concentration is too low they tend to accumulate near the bottom wall at quasi-steady state. On the other hand if the particle concentration is too high, inter-particle collisions keep the particles in suspension resulting in the overall concentration profile being more uniform across the height of channel.
Figure 3.8: Particle position with and without considering inter-particle collision at time $t = 0.5 \text{ secs}$. (Particle sizes are magnified 5 times)
For all simulations involving spherical particles, the size distribution was based on Gaussian distribution function with mean particle diameter equal to 195\(\mu m\) and standard deviation of 21.5\(\mu m\), consistent with the simulations of Mallouppas and van Wachem (2013). Thus the Stokes number of the particles was greater than 1. Particle tracking was started from 35\(T_L\) to 1000\(T_L\) and the data was averaged in time to give a better statistical picture of the concentration and velocity profile of the particles. Here \(T_L\) is the integral time scale of turbulence obtained at the center of the channel (Milojević, 1990).

\[
T_L = \frac{2k}{9\epsilon} \tag{3.12}
\]

\[
k = 0.5(u_{rms}^2 + 2v_{rms}^2) \tag{3.13}
\]

\[
\epsilon = C_{0.75}^k l_{m}^{1.5} \tag{3.14}
\]

where \(k\) is the kinetic energy based on RMS velocity at the center of the channel; \(\epsilon\) is the dissipation rate and was obtained by keeping the coefficient \(C_\mu = 0.09\); \(l_m\) is the mixing length where the ratio of mixing length to channel height was 0.07 (Schlichting, 1965). Kussin and Sommerfeld (2002) reported that \(T_L = 0.0037\) \(secs\) for the case under consideration.

Figure 3.9 shows the concentration and velocity profile obtained by introducing different number of particles in the domain. The concentration of the particles was calculated by dividing the width of the channel into small uniform bins, all having a constant length equal to the length of the channel. The number of particles in each bin were counted at a particular time step and an averaged value over all time steps, which are equal in size, was used to calculate the concentration. The obtained concentration values were normalized with the average concentration obtained for the entire domain. The average
fluid velocities were calculated in a similar manner and then normalized by the bulk fluid velocity, \( u_{\text{bulk}} = 19.7 \text{ m/s} \). It is evident that as the number of particles increased in the domain more particles remain suspended in the flow, lowering the concentration near the bottom wall. For the simulation considering 1500 particles the concentration was so high that the inter-particle collisions were highly frequent. This prevented the particles from settling down under the effect of gravity and thus the concentration profile is more uniform. In comparison to the concentration profile reported by Mallouppas and van Wachem (2013) simulation with 400 particles gives the closest match. Note that the simulations by Mallouppas and van Wachem (2013) are 3-D and fully coupled which results in the discrepancies observed with the concentration profile generated by 2-D EPI code.

The velocities generated by the EPI code for the 2-D simulations were more than what were reported for 3-D simulations by Mallouppas and van Wachem (2013). This can be attributed to the fact that out of plane collisions in 3-D, which are non existent in the current simulations, will decrease the overall kinetic energy of the particles. These out of plane collisions, however, will not affect the width-wise position of the particles statistically. Therefore the concentration profile of 2-D simulations provides a good match with the 3-D data but the velocity profile does not. Some particular observations among the 2-D and 3-d velocity profiles are nevertheless similar. For example both the simulations generate a parabolic type velocity profile for the particles with maximum average velocity near the channel center and minimum near the walls. Further, at the quasi-steady state the particle concentration near the lower wall is high, which means that the inter-particle and particle-wall collisions near the lower face are more frequent. This results in a higher kinetic energy loss for the particles near the lower wall as compared to particles near the upper wall. Therefore the average velocity of the particles near the lower wall is the
lowest in the domain for both 2-D and 3-D simulations.
Figure 3.9: Results obtained from EPI code for spherical particles (considering smooth walls) for different number of particles introduced in domain
3.3.4 Smooth Wall Simulations: Non-spherical Particles

Ellipsoid1, Ellipsoid2 and Fiber as summarized in Table 3.2 were analyzed using the EPI code. Since discs are oblate spheroids (i.e., \( b > c \), where \( b \) and \( c \) are the two minor axis), they are not included in this study. For other particles it is assumed that \( b = c \). From the spherical particle simulations it was established that initializing the solver with 400 particles provided a good match with concentration profile reported by 3-D simulations of Mallouppas and van Wachem (2013). Therefore, all non-spherical particle simulations were also performed using 400 particles to maintain consistency. The particles were tracked from 35 \( T_L \) to 1000 \( T_L \) and the particle concentrations and velocities were calculated in a similar way as for spherical particles. The initial particle orientations were assigned randomly.

For referring to 2-D simulation results the particle will be referred to a Ellipse 1 and Ellipse 2 which correspond to Ellipsoid 1 and Ellipsoid 2 respectively. Figure 3.10 shows the results obtained for Ellipsoid 1 of Table 3.2. The normalized concentration profile of Ellipse 1 is also compared with the spherical particle profile (Figure 3.10a). In contrast to the spherical particles collision of the ellipses with the wall imparts rotational momentum to the particles which is contributed from the pre-collision linear momentum. The resulting post-collision velocity in the wall-normal direction is less as compared to the spherical particles after a collision event. Therefore, the ellipses have a higher tendency of settling on the walls when compared to spherical particles which results in a sharp increase in their concentration near the bottom wall. The EPI code provides a good match to the concentration profile for Ellipsoids 1 in the center of channel. Near the walls the mismatch can be attributed to out-of-plane collisions that further decrease the kinetic
energy of the particles and 'trap' them.

Figure 3.10b shows the average normalized velocity profile of the Ellipse 1 in comparison to spheres and Ellipsoid 1. Similar to spherical particles, the EPI code over predicts the velocities of Ellipse 1 in comparison to Ellipsoid 1. Qualitatively however the results by van Wachem et al. (2015) and the EPI code have some similarities. The highest particle velocity reported by the EPI code is slightly above the center of the channel (around 0.6 normalized height) similar to the Ellipsoid 1 profile. Also, due to the high concentration near the bottom wall, resulting in more frequent collisions and consequently more kinetic energy loss, the average velocity near the bottom wall decreases more sharply as compared to the velocity near the upper wall.

Sphericity of the particle, as listed in Table 3.2, plays an important role in the collision dynamics of the particle. Ellipsoid 2 has a sphericity close to one. The concentration and velocity profiles generated by Ellipsoid 2/ Ellipse 2 are expected to be close to that of spherical particles. However, as reported by van Wachem et al. (2015) both the concentration and velocity profiles for Ellipsoid 2 and spheres are significantly different. Similarly, the reported velocity profile of Ellipsoid 2 is unexpectedly different as compared to the spheres. In the results obtained by the EPI code a slightly higher and lower concentration is observed near the lower wall and upper wall respectively for Ellipse 1 as compared to the respective concentration for the spheres (see Figure 3.11a). The velocity profile by the EPI code for Ellipse 2 closely resembles that for spheres. Naturally, 3-D implementation of the EPI code is required to properly establish the reason for discrepancy between the expected and obtained results. Figure 3.12a shows the concentration profile obtained with the EPI code for Fiber particles. The concentration is slightly over-predicted at the center of the channel at the cost of decrease in concentration near the bottom wall. The
Figure 3.10: Results obtained from EPI code for Ellipse 1, (Aspect Ratio = \( \frac{5}{2} \), sphericity = 0.885) 400 particles (considering smooth walls)
overall trend, however, agrees well with the 3-D simulations. Concentration near the top of the channel matches very well with the results of van Wachem et al. (2015).

A comparison of the concentration and velocity profiles obtained with the EPI code has been presented in Figure 3.13 for all particles of interest. Evidently, with increasing aspect ratio deposition at the lower walls also increases. Clearly, while spherical or near spherical particles tend to bounce off the wall after a collision event, collision of elongated particles imparts rotational energy to the particle at the expense of momentum in the wall normal direction. Thus the elongated particles are susceptible to subsequent collisions resulting in higher concentration at the bottom wall (Figure 3.13a). For the same reason near wall velocity of the particles is also inversely proportional to the aspect ratio, because subsequent collisions lead to higher losses in translational kinetic energy (Figure 3.13b).

Figure 3.14 shows the comparative average angular velocities of the non-spherical particles plotted against the normalized channel height obtained from the EPI code. There are two factors that govern angular velocity of the particles. First is the rotational torque exerted by the surrounding fluid on the particles. The vorticity of the fluid is minimum at the center of the channel and a maximum near the walls. Therefore the rotational torque and hence the angular velocity of the particles increases in magnitude from the center of the channel towards the walls. The moment of inertia of Ellipse 2 is (of the order $\sim 10^{-16} kgm^2$) lowest of all the non-spherical particles. Therefore same amount of torque exerted by the hydrodynamic factors imparts a higher angular velocity to Ellipse 2 particles as compared to other particles. Both Ellipse 1 and Fiber have comparable values for moment of inertia (of the order $\sim 10^{-15} kgm^2$). Therefore their average angular velocities are comparable in most of the domain. The second major factor contributing to the angular velocity of the particles is wall collision. For all particles the gradient of angular velocity profile increases sharply near the walls resulting in a hyperbolic type
Figure 3.11: Results obtained from EPI code for Ellipse 2, (Aspect Ratio = $\frac{5}{4}$, sphericity = 0.991) 400 particles (considering smooth walls)

(a) Concentration profile as a function of dimensionless channel height

(b) Average normalized velocity profile as a function of dimensionless channel height
Figure 3.12: Results obtained from EPI code for Fiber, (Aspect Ratio = 5, sphericity = 0.639) 400 particles (considering smooth walls)
Figure 3.13: Comparison of the results obtained by EPI code for different particle shapes

(a) Concentration profile as a function of dimensionless channel height

(b) Average normalized velocity profile as a function of dimensionless channel height
trend. Note that the trend of angular velocity plots is similar to results reported by Zhao and van Wachem (2013a) for ellipsoidal particles of aspect ratio 3 (see Figure 3.15).

Figure 3.14: Comparison of the average angular velocity of the particles obtained from the EPI code
Figure 3.15: Average angular velocity trend reported by Zhao and van Wachem (2013a) using DNS one-way simulation of 3-D ellipsoid particles with aspect ratio = 3. The profile obtained from the EPI code follows a similar trend.
### Table 3.5: Simulation run times (in clock secs) for different particles with EPI code

<table>
<thead>
<tr>
<th>Name</th>
<th>Shape</th>
<th>Sphericity</th>
<th>Aspect Ratio</th>
<th>Size</th>
<th>Time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sphere</td>
<td>Ⓡ</td>
<td>1</td>
<td>$\frac{a}{b} = 1$</td>
<td>200µm</td>
<td>11367</td>
</tr>
<tr>
<td>Ellipse 1</td>
<td>Ⓢ</td>
<td>0.885</td>
<td>$\frac{a}{b} = \frac{5}{2}$</td>
<td>$a = 368µm$</td>
<td>19108</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$b = 147µm$</td>
<td></td>
</tr>
<tr>
<td>Ellipse 2</td>
<td>Ⓢ</td>
<td>0.991</td>
<td>$\frac{a}{b} = \frac{5}{4}$</td>
<td>$a = 232µm$</td>
<td>14359</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$b = 68.4µm$</td>
<td></td>
</tr>
<tr>
<td>Fiber</td>
<td>Ⓡ</td>
<td>0.639</td>
<td>$\frac{a}{b} = 5$</td>
<td>$a = 510µm$</td>
<td>36987</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$b = 102µm$</td>
<td></td>
</tr>
</tbody>
</table>

#### 3.3.5 Rough-wall Collisions

EPI code was also applied to rough wall collisions. The rough wall algorithm is similar to as applied by Sommerfeld and Huber (1999) and is described in Chapter-2, Section 2.3.2. In the simulations performed by Mallouppas and van Wachem (2013) roughness angle was chosen from a normal distribution with a standard deviation of $\gamma = 5.02^\circ$. In the EPI 2-D code the same value of $\gamma$ generated a concentration profile close to that of smooth wall. Several computations were performed with increasing roughness angles to determine the closest fit to the concentration results generated by van Wachem et al. (2015). For a value of $\gamma = 18^\circ$ the concentration profile of non-spherical particles was found to be a good fit to that of van Wachem et al. (2015). The exact reason why the smooth wall concentration profiles match but an increase in the roughness angle is required to match the concentration profile for rough walls is unknown. A 3-D implementation of the EPI code is required to make concrete comments about rough-wall modelling.
Figures 3.16, 3.17 and 3.18 show the result obtained for Ellipse1, Ellipse2 and Fiber from the EPI code as compared to the profiles reported by 3-D particles by van Wachem et al. (2015) and the results for spheres from the EPI code. It is evident that introduction of rough walls causes re-suspension of the particles after collision resulting in a decrease in average concentration of the particles near the bottom wall. The velocity profiles are over predicted by the EPI code for all particles, similar to the case of smooth walls. A comparison of the results obtained by EPI code for all particles is presented in Figure 3.19. For spherical particles the effect of rough walls is more pronounced as compared to non-spherical particles. This is because after collision with rough wall a larger part of horizontal momentum gets converted to vertical momentum for Sphere and Ellipse 2 (because its sphericity is close to 1) as compared to Ellipse 1 and Fiber. This causes a stronger re-suspension of the spherical particles back into the channel, resulting in their lower concentration near the bottom wall as compared to particles with higher aspect ratio. Because the horizontal momentum for elongated particles (Ellipse 1 and Fiber) is relatively preserved after collision, they have higher average near wall velocities as compared to the spherical particles. A more uniform concentration profile for spherical particle in the channel results in higher inter-particle collision in the channel center and consequently lower kinetic energies. Therefore the maximum average velocity of the Sphere particles is the lowest and of Fiber the highest.
Figure 3.16: Results obtained from EPI code for Ellipse 1, (Aspect Ratio = \( \frac{5}{2} \), sphericity = 0.885) 400 particles (considering rough walls, \( \gamma = 18^\circ \))
Figure 3.17: Results obtained from EPI code for Ellipse 2, (Aspect Ratio = $5/4$, sphericity = 0.991) 400 particles (considering rough walls, $\gamma = 18^\circ$)
Figure 3.18: Results obtained from EPI code for Fiber, (Aspect Ratio = 5, sphericity = 0.639) 400 particles (considering rough walls, $\gamma = 18^\circ$)
(a) Concentration profile as a function of dimensionless channel height

(b) Average normalized velocity profile as a function of dimensionless channel height

Figure 3.19: Comparison of the results obtained by EPI code for different particle shapes (considering rough walls, $\gamma = 18^\circ$)
3.4 Chapter Summary

This chapter focuses on the application of the theory developed in Chapter-2 for modelling ellipsoidal collision. The EPI model based on the so called ‘psuedo-sphere’ approach is applied to solve for the transport of some specific ellipsoidal particles of aspect ratios $5/2$, $5/4$ and $5$ and also spherical particles. Again, van Wachem et al. (2015) is the chosen benchmark study for comparison which analyzed the same non-spherical particles using fully coupled 3-D simulations. Major differences in the present study and the benchmark study are that the present simulations are one-way coupled, i.e., the particles do not affect the flow field and the collisions and the angular velocity of the particles are constrained to only one plane. Still a comparison can be made out between the current results and the results by van Wachem et al. (2015) since they employed periodic boundary conditions in order to effectively simulate flow between infinite parallel plates.

Concentration profile results obtained from the EPI code match well with the benchmark study. The average velocity predicted by the EPI code however exceeds the LES results. This has been argued to be attributed to the effects of out-of-plane collisions which do not change the wall normal position of the particle but does decrease the kinetic energy. Comparison of the results obtained for different particles obtained with the EPI code and the benchmark study show consistencies. The particle concentrations and velocities near the walls are a function of the aspect ratio of the particles. Fibers, with the highest aspect ratio have the highest concentration and the lowest velocities near the bottom wall, whereas Ellipsoid 2, with sphericity close to 1, has the lowest concentration and highest velocities.

In summary, The EPI model can be viewed as a viable approach for non-spherical particle collision modelling.
4.1 Conclusions

Particle-laden flows are an area of important research activities. Several factors need to be considered while modelling the transport of particles in a fluid flow e.g., mapping Eulerian fields to Lagrangian phase, turbulence modification of the Eulerian phase due to the presence of particles, hydrodynamic forces and torques acting on the particle, etc. However in applications where the particle position and velocity are of prime importance, perhaps the most important factor to be accounted for is inter-particle and particle-wall collision, since they have maximum impact on the trajectory of particles and hence deposition. Thus accurate and efficient modelling of collision events is crucial to particle-laden flows. Based on the literature review the following conclusions can be drawn.

1. For non-spherical collision modelling the most widely used method both in industry and academia is the multi-sphere (MS) method. It is popular because it is easy to implement in computer programs and provides rigorous collision detection and force modelling mechanisms.
2. Validity of the MS approach has not been established conclusively. Conflicting studies exist in literature which have attempted to prove the applicability of MS method as a proper representation of non-spherical (or even spherical) particles.

2.1. Kruggel-Emden et al. (2008) studied the bouncing of spherical particles off a wall where the particle was modelled using MS method. They observed large errors in the bouncing characteristics with the MS approach and also showed that there was no correlation between the number of comprising sub-spheres used to model the particle and the accuracy obtained. They however stated that MS method may be applied to granular flows where the particle shapes are truly random but the MS method does not accurately simulate single grain collision characteristics.

2.2. Song et al. (2006) carried out simulations using the MS method on collision of tablets. An exact algorithm specifically developed for tablet shapes was used for comparison. Their results showed that the MS method yielded very different results from the exact algorithm. Further using 66-178 spherical particles to represent the tablet, the simulation run times were infact more as compared to exact algorithm.

2.3. Kačianauskas et al. (2014) implemented the MS model to represent particles with randomly shaped surfaces. They concluded that the dependency of impact characteristics on MS refinement was non-unique.

2.4. Markauskas et al. (2010) investigated the applicability of MS method to ellipsoids by simulating piling of ellipsoid particles. The 3-D ellipsoids modelled using MS method were compared with 2-D ellipses modelled analytically. They concluded that the MS method was successful in solving the piling prob-
lwm with good accuracy. (Garcia et al., 2009; Ferellec and McDowell, 2010; González-Montellano et al., 2012)

2.5. Abbaspour-Fard (2004) studied normal collisions of elongated particles modelled using MS approach and compared it to analytical values. Their study supported the MS approach showing good agreement between the computational and analytical results.

From the above listed studies we can conclude that the investigations which were focused on the single grain bouncing characteristics did not support the MS model. However the studies that applied the MS model to granular type flow or piling simulations supported it. More studies on granular flow simulations using MS method include Garcia et al. (2009), Ferellec and McDowell (2010), González-Montellano et al. (2012), etc.

3. As reported by Markauskas et al. (2010) the simulation time required for non-spherical particles increases non-linearly with an increase in number of comprising sub-spheres.

The focus of this study is on particle-laden flows only, where the particles are not in constant contact with each other as in granular flows. Thus the impact characteristics of individual particles are important. A novel modelling approach for ellipsoidal particle interaction (EPI) has been developed and implemented in 2-D turbulent channel flow. The results obtained are compared with existing computational data for 3-D ellipsoids. With respect to the result comparison with the LES benchmark computational data of van Wachem et al. (2015), the following conclusions can be drawn.

1. Concentration profiles obtained with the EPI code agrees well with the 3-D simulation data for both spherical and non-spherical data.
2. Particle velocities are over predicted using the EPI code which can be attributed to kinetic energy losses in 3-D simulations due to out-of-plane collisions. Nevertheless there are certain similarities in the velocity profiles of 2-D and 3-D particles.

With respect to the simulations with the EPI model the following qualitative observations can be made:

1. Collision detection of the ellipsoids is based on the exact analytical equations. Thus, it is subject to machine round-off errors only.

2. Contact point calculation is based on an algorithm similar to geometric potential algorithm (GPA) (Lin and Ng, 1995). The GPA was reported to calculate the collision point with high accuracy and efficiency. Discussion on the collision point by Lu et al. (2015) as applied to the MS method suggests that the inaccuracy in impact characteristics for a particle represented by sphere agglomerates is mainly due to the existence of multiple contact points which are possible in the collisions using the MS approach. For the EPI model, however, only one contact point exists which is itself calculated from analytic equations of ellipsoids.

3. Collision force and torque calculation as a part of the EPI model is similar to the MS method i.e., it is based on pseudo sub-spheres. The major difference however is that the EPI model is designed such that the ellipsoid can be imagined to be composed of an infinite number of sub-spheres. Thus the resulting forces and moments from the EPI model are expected to be more accurate than the MS approach.

4. A single run of the collision detection and contact point calculation routines are expensive for the EPI method when compared to the MS method. The MS method however is more expensive in terms of the computer memory requirements. Further
the near-neighbor lists (NNL) generated in the MS method will be $N^2$ times bigger for a particle pair as compared to the NNL in EPI, where $N$ is the number of comprising sub-sphere in each particle. This consequently increases both the memory requirements and the expense of running contact detection for $N^2$ pairs. Thus for a smaller number of particles being simulated the MS method might be more efficient, however, as the particle increase in the system the EPI code may prove to be less expensive.

5. Based on the results obtained for the channel flow simulation using the EPI model, it is suitable to be adopted as a viable model for 3-D ellipsoid collision. 3-D implementation of the EPI model is required to present a more rigorous comparison with the MS method in terms of accuracy and efficiency.

### 4.2 Future Work

The EPI algorithm is developed with the objective of studying non-spherical particle transport in laminar and turbulent flow fields in complex flow geometries occurring in applications such as therapeutic aerosol transport in lung geometries, nano-fluid cooling applications, micron-fibers used in cancer treatment etc., all relying highly on non-spherical particle dynamics. Figure 4.1 illustrates the proposed steps in the development and application process of the EPI model.
4.2.1 3-D Implementation of EPI Model

The first step to proceeding with the modelling of non-spherical particles is the extension of EPI model to 3-D collisions. The following sections describe the theory of 3-D EPI modelling.

4.2.1.1 Collision Detection for 3-D Ellipsoids

Prior to the analysis by Wang et al. (2001) on collision detection of ellipsoids, there were several studies carried out. Farouki et al. (1989), Levin (1979) and Wilf and Manor (1993)
provided methods for finding the intersection of any quadric surfaces and hence can also be applied to intersection of ellipsoids. These algorithms were however extremely complex and are expensive to be adopted in programs simulating a large number of particles. Numerical algorithms were provided by Perram and Wertheim (1985), Perram et al. (1996) and Lin and Ng (1995) which were iterative and thus computationally expensive. The method described by Wang et al. (2001) gives an algebraic condition for determining whether two ellipsoids intersect or not. The method is thus concise and exact. Similar to the explanation in Chapter-2, Section 2.3.1.1 the equation of two ellipsoids can be written in matrix form as,

\[ X^TAX = 0 \]  \hspace{1cm} (4.1)
\[ X^TBX = 0 \]  \hspace{1cm} (4.2)

where \( X = (x, y, z, 1)^T \) and \( A \) and \( B \) are the coefficient matrices of the two ellipsoids. Through affine transformation one of the ellipsoid is converted to canonical form and the other is converted to a sphere. Thus if \( T \) is the transformation matrix the new coefficient matrices are given by,

\[ \tilde{A} = (T^{-1})^TAT^{-1} \]  \hspace{1cm} (4.3)
\[ \tilde{B} = (T^{-1})^TBT^{-1} \]  \hspace{1cm} (4.4)

The characteristic equation of the original ellipsoids is,

\[ f(\lambda) = \det(\lambda A + B) = 0 \]  \hspace{1cm} (4.5)
The characteristic equation of the transformed ellipsoids is given by,

\[ f(\lambda) = det(\lambda \tilde{A} + \tilde{B}) = det^{-2}(T)f(\lambda) = 0 \]  

(4.6)

From Equation 4.6 the roots of the two characteristic equations are the same. Therefore the relationship between the ellipsoids will not change under affine transformation. If the coefficient matrices of the transformed ellipsoids are given by,

\[
\tilde{A} = \begin{bmatrix}
\frac{1}{a^2} & \frac{1}{b^2} & \frac{1}{c^2} \\
\frac{1}{b^2} & \frac{1}{c^2} & -1
\end{bmatrix} \tag{4.7}
\]

\[
\tilde{B} = \begin{bmatrix}
1 & -x_c & -x_c & -r^2 + x_c^2 + y_c^2 + z_c^2 \\
-1 & 1 & -y_c & -z_c \\
-1 & -1 & 1 & -z_c \\
-x_c & -y_c & -z_c & 1
\end{bmatrix} \tag{4.8}
\]

where \( \tilde{A} \) is the transformed ellipsoid centered at origin and \( \tilde{B} \) is the transformed sphere with radius \( r \). The characteristic polynomial of the conics will be,

\[
f(\lambda) = -\left( \frac{\lambda}{a^2} + 1 \right) \left( \frac{\lambda}{b^2} + 1 \right) \left( \frac{\lambda}{c^2} + 1 \right) (\lambda + r^2) + \frac{x_c^2}{a^2} \left( \frac{\lambda}{a^2} + 1 \right) \left( \frac{\lambda}{b^2} + 1 \right) \lambda + \frac{y_c^2}{b^2} \left( \frac{\lambda}{a^2} + 1 \right) \left( \frac{\lambda}{c^2} + 1 \right) \lambda + \frac{z_c^2}{c^2} \left( \frac{\lambda}{a^2} + 1 \right) \left( \frac{\lambda}{b^2} + 1 \right) \lambda \tag{4.9}
\]

Equation 4.9 is a fourth order polynomial in \( \lambda \) and thus will have 4 roots. The sign of the roots of this equation determine the relative position of ellipsoids. Wang et al.
(2001) proved that the ellipsoids are separated by a plane only if two of the roots of the characteristic equations are distinct and positive.

Similar to the approach adopted by Etayo et al. (2006) for characterizing the relative position of two ellipses using a set of 4 polynomials (see Chapter-2, Section 2.3.1.1), Gonzalez-Vega and Mainar (2008) provided relations for determining the relative position of ellipsoids based on a set of 6 polynomials. Their method is a result of the algebraic condition put forth by Wang et al. (2001) but provides a more convenient way of contact detection using the coefficient matrix of two ellipses.

An alternative method for contact detection, which is also used for contact point calculation of two ellipsoids is described in the next section. A comparison of the two methods must be made via computer implementation to determine the most suitable method in terms of accuracy and efficiency.

### 4.2.1.2 Calculation of Collision Point of Ellipsoids

Geometric potential algorithm (GPA) was introduced by Lin and Ng (1995) to determine the collision point for two overlapping ellipsoids. It has been compared with the intersection algorithm and common normal algorithm and it has been established that it is the most accurate and robust of the three available methods (Džiugys and Peters, 2001b). Let $A$ and $B$ be general ellipsoids whose equations are given in the general polynomial form as,

$$A = a_1 x^2 + b_1 y^2 + c_1 z^2 + 2f_1 yz + 2g_1 zx + 2h_1 xy + 2p_1 x + 2q_1 y + 2r_1 z + d_1 = 0 \quad (4.10)$$

$$B = a_2 x^2 + b_2 y^2 + c_2 z^2 + 2f_2 yz + 2g_2 zx + 2h_2 xy + 2p_2 x + 2q_2 y + 2r_2 z + d_2 = 0 \quad (4.11)$$
The fundamental behind the GPA is to find a point on the surface of ellipsoid $A$ which is deepest inside the other ellipsoid $B$. Thus a potential function is constructed based on a parameter $\lambda$ subject to constraint that $A = 0$

$$\Pi = B + \lambda A$$

(4.12)

minimizing the potential yields,

$$\begin{bmatrix}
  a_2 + \lambda a_1 & h_2 + \lambda h_1 & g_2 + \lambda g_1 \\
  h_2 + \lambda h_1 & b_2 + \lambda b_1 & f_2 + \lambda f_1 \\
  g_2 + \lambda g_1 & f_2 + \lambda f_1 & c_2 + \lambda c_1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix}
= -
\begin{bmatrix}
  p_2 + \lambda p_1 \\
  q_2 + \lambda q_1 \\
  r_2 + \lambda r_1
\end{bmatrix}$$

(4.13)

In the above equation the unknowns are $x, y, z$ which can be evaluated in terms of $\lambda$ using Cramer’s rule,

$$x = \frac{D_1}{D}$$

(4.14)

$$y = \frac{D_2}{D}$$

(4.15)

$$z = \frac{D_3}{D}$$

(4.16)

Equation 4.16 can now be substituted back into the equation of Ellipse $A$ to obtain a sixth order polynomial

$$a_1D_1^2 + b_1D_2^2 + c_1D_3^2 + 2f_1D_2D_3 + 2g_1D_3D_1 + 2h_1D_1D_2$$

$$+ 2p_1D_1D + 2q_1D_2D + 2r_1D_3D + d_1D^2 = 0$$

(4.17)

This equation can be solved numerically via methods such as the Newton-Raphson
method to obtain real values of $\lambda$ which minimize the potential function. Finally the coordinates of the desired point may be obtained by substituting the obtained value of $\lambda$ back into equation 4.13. The process is repeated to obtain the point on Ellipsoid $B$ which is deepest inside $A$. The collision point is chosen to be the midpoint of line joining the two deepest points.

### 4.2.1.3 Calculation of interacting Pseudo-sphere

After the collision point has been calculated the calculation of the center of interacting pseudo-sphere is similar to 2-D as explained in Chapter-2, Section 2.3.3.2. The following algorithm is proposed (see Figure 4.2),

1. The parametric equation of ellipsoid in spherical coordinates, with respect to a coordinate system fitted at the center of ellipsoid, is

   \[
   x = a \cos(\phi) \cos(\theta) \\
   y = b \cos(\phi) \sin(\theta) \\
   z = c \sin(\phi)
   \] (4.18) (4.19) (4.20)

   From the knowledge of the collision point the angle $\phi$ can be calculated.

2. Transform the coordinate system by rotating the body fitted coordinate system by an angle $\phi$ about the original $z$ axis. In this new coordinate system the ellipsoid
can be collapsed to an ellipse with the parametric equation,

\[ x' = a \cos(\theta) \]  \hspace{1cm} (4.21) \\
\[ y' = b \sin(\theta) \]  \hspace{1cm} (4.22)

where \( x' \) and \( y' \) are the new coordinates. Let \( x'_c \) and \( y'_c \) be the position coordinates of the collision point in the new coordinate system.

3. A pencil of lines passing through the collision point is defined as

\[ y' = y'_c + m(x' - x'_c) \]  \hspace{1cm} (4.23)

Intersection points are calculated in terms of \( m \) by solving 4.23 simultaneously with equation of ellipse. The line segment obtained by the resulting points is minimized in terms of \( m \) to obtain the slope. The plane obtained by converting this line segment to 3-D has the minimum cross-sectional area with the ellipsoid and is selected to be the collision plane for both interacting ellipsoids. Note that for real particles being simulated by DEM the overlap area is very small. Thus this plane will serve as a good approximation of the actual collision plane.

4. The radius of the interacting pseudo-sphere for each ellipsoid will be given by the line segment passing through the collision point and meeting the major axis of ellipsoid, perpendicular to the collision plane.
Figure 4.2: Description of the algorithm for calculating the interacting pseudo-sphere for 3-D ellipsoid collision
REFERENCES


Hamilton, W. R. (1844). Lxxviii. on quaternions; or on a new system of imaginaries in algebra: To the editors of the philosophical magazine and journal.


139


