Timo Niemi

Particle Size Distribution in CFD Simulation of Gas-Particle Flows

School of Science

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 5.6.2012

Thesis supervisor:

Prof. Rolf Stenberg

Thesis instructor:

Lic. (Tech.) Sirpa Kallio

Aalto University
School of Science

AALTO UNIVERSITY
SCHOOL OF SCIENCE

ABSTRACT OF THE
MASTER'S THESIS

Author: Timo Niemi

Title: Particle Size Distribution in CFD Simulation of Gas-Particle Flows

Date: 5.6.2012 Language: English Number of pages:8+84

Department of Mathematics and Systems Analysis

Professorship: Mechanics Code: Mat-5

Supervisor: Prof. Rolf Stenberg

Instructor: Lic. (Tech.) Sirpa Kallio

Fluidized bed combustion (FBC) boilers have become one of the leading technologies in environment-friendly biomass combustion. In fluidized bed combustors the solid fuel particles are suspended on upward-blowing air resulting in a turbulent mixing of gas and solids. This mixing process allows efficient chemical reactions and heat transfer which in turn help to reduce emissions and allow to utilize many different types of fuels.

In order to develop even better FBC designs, numerical simulations could be used to model the flow behaviour inside the boilers. However, the complex nature of the multiphase flow of particles and combustion air makes the modelling very challenging. One of the issues that requires consideration is the size distribution of the particles. Traditionally only the average size of the particles has been used in the simulations in order to keep them simpler but in reality the size distribution also affects the flow behaviour and should be taken into account.

In this thesis the relatively recently introduced approaches for particle size distribution (PSD) modelling in CFD setting are studied. The applicability of the approaches to fluidized bed simulation are examined and the computational requirements of the approaches are compared. Based on the these factors, currently the moment methods and of these especially the DQMOM-method appear to be most suitable for fluidized bed simulations.

As many of the studied approaches can increase the computational requirements of the simulations substantially, a new, simplified method to model the PSD is also introduced in this thesis. The new method is tested by simulating a small scale fluidized bed and the results are compared to measurements and to the results obtained from simulations with a single particle size.

Keywords: particle size distribution, fluidized bed, CFB, multiphase, DQMOM, Euler-Euler, CFD

AALTO-YLIOPISTO
PERUSTIETEIDEN KORKEAKOULU

DIPLOMITYÖN
TIIVISTELMÄ

Tekijä: Timo Niemi

Työn nimi: Partikkelikokojakauman huomioiminen
kaasu-partikkelivirtausten simuloinnissa.

Päivämäärä: 5.6.2012 Kieli: Englanti Sivumäärä:8+84

Matematiikan ja systeemianalyysin laitos

Professuuri: Mekaniikka Koodi: Mat-5

Valvoja: Prof. Rolf Stenberg

Ohjaaja: TkL Sirpa Kallio

Leijupetikattilat ovat yksi tärkeimmistä biopolttoaineille soveltuvista kattilatyypeistä. Leijupetikattiloissa polttoainepartikkelit leijuvat alhaalta päin tulevan ilmavirtauksen varassa, mikä johtaa polttoaineen ja ilman tehokkaaseen sekoittumiseen. Hyvä sekoittuminen tasaa lämpötiloja kattilassa ja mahdollistaa päästöjen vähenemisen ja monipuolisen polttoainevalikoiman käytön.

Leijupetikattiloiden kehityksen nopeuttamiseksi olisi hyödyllistä, jos kattiloissa tapahtuvaa polttoprosessia voitaisiin mallintaa numeerisesti. Partikkeleiden ja polttoilman muodostama monimutkainen monifaasivirtaus on kuitenkin hyvin haasteellinen mallinnettava ja laskentamenetelmiä on edelleen kehitettävä. Eräs ongelma, joka mallinnukseen liittyy on partikkeleiden kokojakauman huomioon ottaminen. Perinteisesti partikkelit on mallinnettu pelkästään niiden keskikoon avulla mallinnuksen yksinkertaistamiseksi, mutta myös kokojakauma tulisi ottaa huomioon, koska se vaikuttaa virtauksen käyttäytymiseen.

Tässä diplomityössä tutkitaan viime aikoina kehitettyjä menetelmiä partikkelikokojakauman mallintamiseen monifaasivirtauslaskennassa. Pääasiallisena vertailukohtana eri menetelmien välillä käytetään sekä niiden soveltuvuutta leijupetien mallintamiseen että menetelmien vaatimaa laskentatyön määrää. Vertailun perusteella momenttimenetelmät ja näistä erityisesti DQMOM-menetelmä vaikuttaa tällä hetkellä soveltuvimmalta lähestymistavalta.

Koska monet tutkituista menetelmistä lisäävät tarvittavan laskentatyön määrää merkittävästi, tässä työssä kehitetään myös vaihtoehtoinen, kevyt menetelmä kokojakauman mallintamiseen. Uutta menetelmää testataan simuloimalla pienen kokoluokan leijupetiä ja saatuja tuloksia verrataan mittauksiin ja yhdellä partikkelikoolla laskettuihin tuloksiin.

Avainsanat: partikkelikokojakauma, leijupeti, CFB, monifaasi, DQMOM,
Euler-Euler, CFD

Preface

This Master's thesis has been written at VTT Technical Research Centre of Finland as a part of Tekes project: CFD based on-line process analysis - applied to circulating and bubbling fluidized bed processes (OnlineFB-CFD). I gratefully acknowledge the financial support of Tekes and all the industrial partners which are involved in the project.

I would like to thank Sirpa Kallio for her excellent guidance and feedback throughout this work and for introducing me to the world of multiphase flow modelling. I am also grateful to Juho Peltola and other colleagues here at VTT for sharing their knowledge in many aspects related to this work. The experimental measurements used in this thesis were conducted by Markus Honkanen. Alf Hermanson helped to operate the experimental devices and has constructed much of the experimental setup.

I would like to thank my supervisor Rolf Stenberg for proof-reading this thesis and for the general guidance during my studies.

Finally, I would also like to thank my parents for their encouragement, my friends for bringing balance to work and especially I would like to thank Ella, for her love and support.

Espoo, 11.5.2012

Timo Niemi

Contents

Symbols and abbreviations

Latin Symbols

A	Area
C_{D}	Drag coefficient
d_{s}	Particle diameter
e_{ss}	Restitution coefficient
f	Number density function
$\mathbf{g}$	Gravitational acceleration
$g_{0,\mathrm{ss}}$	Radial distribution function
I	Identity matrix
$I_{2\mathrm{D}}$	Second invariant of the deviatoric stress tensor
K_{gs}	Interphase drag coefficient
L	Particle length, ie. particle diameter
p	Pressure
Re	Reynold's number
S	Source term
U_{mf}	Minimum fluidization velocity
V	Volume
$\mathbf{v}$	Velocity vector
w	Quadrature weight

Greek Symbols

α	Volume fraction
δ	The Dirac delta function
δ_{qs}	Kronecker delta
ε	Voidage
λ	Bulk viscosity
μ	Viscosity
ϕ	Sphericity
ϕ_{f}	Angle of internal friction
ρ	Density
τ	Stress tensor
Θ	Granular temperature
ξ	Internal coordinate, quadrature abscissa

Subscripts

col	Collisional
f	Fluid
fr	Frictional
g	Gas
kin	Kinetic
ktgf	Kinetic Theory of Granular Flow
mf	Minimum fluidization state
p	Particle
q	Phase indicator
s	Solid, solid mixture
t	Terminal

Abbreviations

BFB	Bubbling Fluidized Bed
CFB	Circulating Fluidized Bed
CFD	Computational Fluid Dynamics
CM	Class methods
CPFD	Computational Particle Fluid Dynamic
DEM	Discrete Element Method
DPM	Discrete Particle Model
DQMOM	Direct Quadrature Method Of Moments
FBC	Fluidized Bed Combustion
KTGF	Kinetic Theory of Granular Flow
MOM	Method Of Moments
MP-PIC	Multiphase Particle-in-Cell method
MUSIG	Multiple Size Group model
N-S	Navier-Stokes (equations)
PBE	Population Balance Equation
PD	Product-Difference (algorithm)
PSD	Particle Size Distribution
QMOM	Quadrature Method Of Moments
SMD	Sauter Mean Diameter
TFM	Two Fluid Model
UDF	User Defined Function
VOF	Volume Fraction

1 Introduction

1.1 Background

Since their introduction in the 1920s, fluidized bed processes have become an important and widely used technology in chemical and metallurgical industries as well as in power generation. These industrial processes are typically large and complex and the prototyping and development of new systems is expensive and time consuming. In order to make the development process more efficient, computational simulation of fluidization has gained interest. The computational fluid dynamics (CFD) approach has emerged as a popular tool in assisting the development of fluidized bed processes.

The flows in fluidized beds are inherently multiphase consisting of at least one fluid phase and one particulate solid phase. The CFD modelling of single phase flows is already quite a challenging task and the multiphase aspect only adds in difficulty. The modelling of dense particulate flows is especially complicated since the flow behaviour in these cases is quite different from the ordinary fluid flows. These challenges make CFD simulation less straightforward to apply, but with proper simulation models good results can still be achieved. However, careful validation of the models and comparisons with experimental data are still very important.

One of the key problems in particulate flow modelling is the particle size distribution (PSD). In virtually all real life situations the particles in fluidized beds are polydisperse. Typically also the chemical reactions and physical processes inside the reactors constantly affect the sizes of the particles. The shape and the properties of the particle size distribution can have dramatic impact on the behaviour of the particulate flow. Thus, in order to accurately model a fluidization process, the particle size distribution and the size change mechanisms have to be taken into account in the computational models.

Traditionally the particle phase has been modelled by using only one, mean particle size disregarding other important characteristics of the particle size distribution. This has been done for efficiency reasons as the modelling of the size distribution can be computationally intensive. At present the increased computational power and improved modelling methods have made more accurate PSD modelling viable. The subject is, however, still under active research and there are various alternative approaches available. In this master's thesis the issue of modelling particle size distributions in gas-particle flows is studied.

1.2 Goals and Objectives

The goal of this thesis is to implement a method for fluid dynamic modelling of particle size distribution in a fluidized bed reactor. Different methods available for modelling PSDs will be discussed and the most promising method will be used as a basis for the implementation. The goal is to have a method, which is capable of:

- working as a sub-model within existing computational fluid dynamics solvers
- modelling particle size distribution with using only reasonable amount of extra computation
- working with dense gas-particle flows

The chosen method will be implemented to work with commercial CFD solver ANSYS FLUENT [1]. The implementation of the method in this thesis is limited to two dimensional modelling. Also the various mechanisms for particle size changes, such as breakage and agglomeration, are left out of scope. However, the implemented method should be readily extensible to 3D modelling and it should include support for extending the implementation with size transfer mechanism later on.

The implementation will be tested by simulating a laboratory scale cold model of a circulating fluidized bed. The simulation results will be compared to other simulations and experimental data available.

1.3 Structure of the Thesis

In chapter 2, the general physical phenomenon of fluidization is described. Also some historical and current applications of fluidization are presented. In chapter 3 the general theory of multiphase flow simulation and the various alternatives of modelling are discussed. Chapter 4 presents different methods for modelling particle size distribution within the CFD framework.

Chapter 5 presents the implementation details of the simulation model chosen. The simulation results and comparisons with experimental data are given in chapter 6. Finally in chapter 7 the conclusions and further work on this subject are discussed.

2 Background

2.1 The fluidization phenomena

Fluidization is a phenomenon where a granular material, such as sand, transforms from a solid-like state into a fluid-like state. Fluidization occurs when a fluid, liquid or gas, is passed up through the granular material. Depending on the flow rate, the properties of the particles and the type of the fluid several different states of fluidization are possible. Figure 1 describes a situation where fluid is passed upwards through a bed of fine particles at different flow rates. If the flow rate is low, the fluid merely flows through the empty spaces between the particles and the bed remains stationary. This situation is called fixed bed. If the flow speed is increased, the drag forces between the fluid and the particles become larger and the bed begins to expand in volume. Eventually a limit is reached where the drag force is in balance with the gravitational force and the particles are suspended within the flow. At this point the bed is considered to be at minimum fluidization state and the particles start to exhibit fluid-like behaviour. [2]

Up to the minimum fluidization point both liquid-solid and gas-solid systems have a relatively similar behaviour. However, when the flow speed is further increased the differences between the two types of systems become substantial. In the liquid-solid case the bed typically simply expands smoothly when the flow rate is increased. The flow stays relatively homogeneous and there are no large variations in the particle concentration. On the other hand, in gas-solid systems the excess gas usually starts to form channels and bubbles within the bed. The bubbles are formed near the gas entry points at the bottom and they rise up through the bed finally bursting up when they reach the bed surface. On the way up the bubbles change in size and shape as they collide with other bubbles or break up into smaller bubbles. If the bed is relatively tall and narrow, the bubbles can fill up the entire width of the bed and the flow becomes slugged.

One way to visualize a bubbling bed is to compare it to boiling liquid. Besides the general appearance, a gas fluidized bed will behave like a liquid in many ways. For example, the bed will conform to the shape of its container and if the container is tilted, the bed's surface tends to remain horizontal. Also if a light object is dropped into a fluidized bed it can rise up and float on the surface. When two containers with different heights of beds are connected, the surfaces of the beds try to equalize. The liquid-like behaviour also allows one to convey the solids from one container to another, using pipes in similar fashion as with liquids.

If one continues to increase the flow speed beyond what is needed for bubbling flow, the nature of the particle bed changes again. The bed height increases and its surface becomes less clearly defined. The bubble shapes become more and more distorted and quite suddenly instead of bubbling flow, one can observe very complex, turbulent-like motion of particle strands and clusters. This state is known as turbulent fluidization and in this regime the mixing of particles is very vigorous.

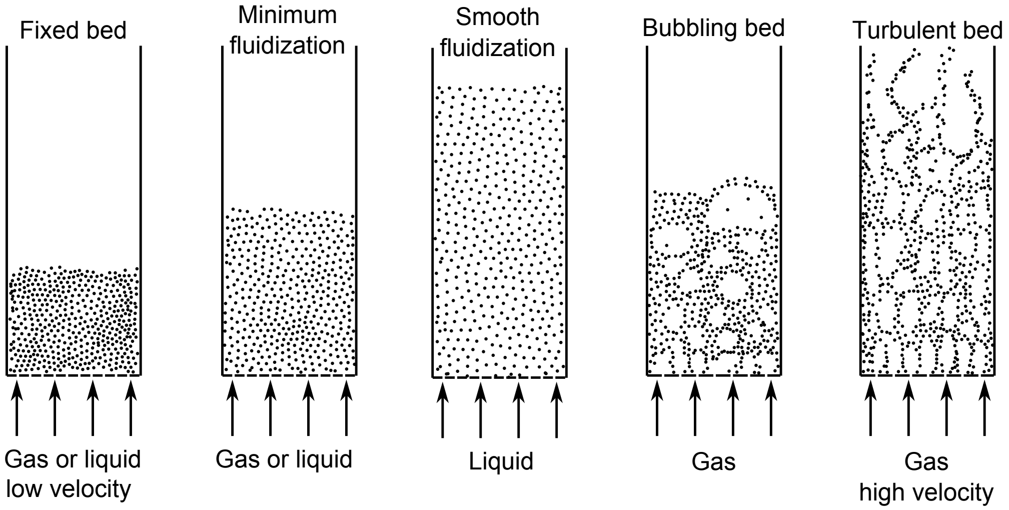

Figure 1: Different states of fluidization.

With even higher flow velocities than what is required for turbulent fluidization a substantial amount of the particles start to leave the bed as they become entrained in the up-going flow. Stable operation at this point requires either constant supply of new particles or a recirculation path for the escaped particles. At this point the behaviour of the bed can be controlled by adjusting the feed rate of particles. If the feed rate is kept relatively high, so that the particles are not completely blown away, the bed is called a fast fluidized bed. A fast fluidized bed typically consists of a relatively dense, turbulent bottom bed which gets more dilute higher up. The dilute region contains particle clusters which are rapidly breaking apart and reforming.

The excellent mixing and good contact between the fluid and particle phases makes fast fluidization interesting for many industrial applications. A fast fluidized bed with a return channel is called a circulating fluidized bed (CFB) and figure 2 shows a simplified example of what it can look like. The particles are fluidized by fast, up blowing air in the riser section. Some amount of the particles reach the top of the riser, where they are separated from the up flowing gas in a cyclone. From the cyclone the particles are dropped to the return channel. At the bottom of the channel there is a loop seal which prevents the fluidization air entering the return channel. The particles are circulated in this fashion until the desired reactions have been achieved. Of course the diagram in figure 2 is much simplified and more complex processes may even require several reactors working in parallel. The basic idea however remains the same. Many industrial processes also use fluidization in the bubbling regime and these are called bubbling fluidized beds (BFB).

Finally, if flow rate is high enough and if the particle concentration is kept very low, ie. the volume fraction < 1%, the situation is called pneumatic transport. Under pneumatic transport there is practically no interaction between the particles. Pneumatic transport can be used for instance to transport solids from one container to another.

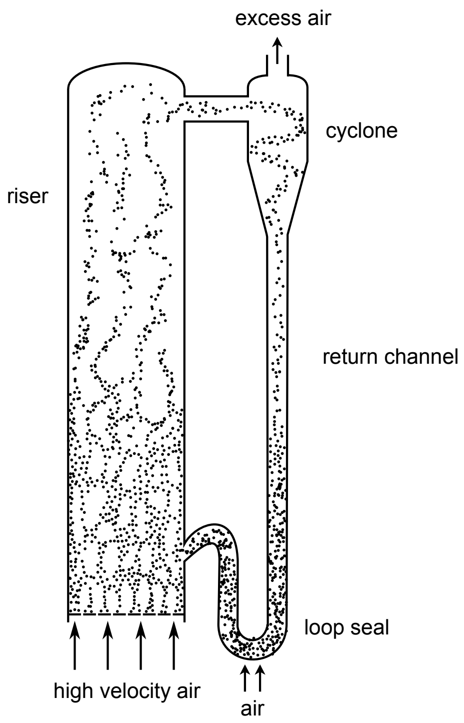

Figure 2: Simplified diagram of a circulating fluidized bed.

2.2 Applications of fluidization

The history of fluidized bed applications begins in the 1920s, when the Winkler process was developed. The process used a fluidized bed to gasify coal with oxygen. However, these first fluidized bed reactors were not very efficient and the real breakthrough of fluidization had to wait until the 1940s. At that time, the war effort demanded ever increasing supply of good quality fuel and the existing refinement processes were not sufficient. This led to the development of the fluid catalytic cracking (FCC) process, which is a method of oil cracking based on fluidized catalysts. The first commercial FCC unit went into operation in 1942 and it was an instant success. The FCC process proved to be much more effective than other techniques available and even today, the FCC process is one of the most important methods to crack crude oil in modern oil refineries. [2]

After the success story in oil refinement, fluidization technology has spread to many different application areas. For example, fluidized beds are used in granulation, drying and coating of substances in various industrial processes ranging from metallurgical to food and pharmaceutical processes. Fluidization is also used in the production of many types of industrial polymers, one notable example being polyethylene, which is the most used plastic in the world. Since the fluidization technology is still relatively recent innovation, new application areas are being actively researched. For example, currently the most promising large-scale production method for carbon nanotubes is the chemical vapor deposition which is based on fluidized bed reactors [3].

Nowadays perhaps one of the most interesting application fields for fluidization is the fluidized bed combustion (FBC). A FBC boiler is a fluidized bed reactor, where the fuel is inserted and burned inside a bed of fluidized particles. Typically the primary fluidizing air is brought from the bottom of the reactor and on the side walls there are fuel injectors and additional air injectors. The boilers can either be BFB reactors or CFB reactors depending on the size and type of usage. The bed material can be for example sand.

The FBC boilers were introduced in 1970s and they are currently gaining popularity because they have many advantages over the traditional combustion boilers, such as grate firing and pulverized coal (PC) boilers. The first key advantage is the lower combustion temperature inside the FBC boilers. The good mixing property of the fluidized bed prevents large temperature gradients from occurring and thus there are few hot spots inside the reactor. The large bed mass also acts as a thermal flywheel enabling stable operation at a certain temperature. The lower combustion temperature is an advantage, because the formation of NO_x emissions is largely temperature dependent. With low enough combustion temperatures it is possible to substantially reduce the NO_x emissions. Also, because of the low temperatures, there is less problems with ash melting. [4]

Another important advantage of FBC reactors is their ability to burn many types of fuels with varying quality. The traditional PC boilers require that the fuel particles have to be ground into very fine powder. A FBC boiler can operate with more coarse particles and with wider size distribution thus reducing the energy spent on grinding. By adding sulphur-absorbing chemical, such as limestone or dolomite to the bed material, it is possible to cheaply reduce SO_x emissions. This allows one to use low quality fuel material, which would otherwise be expensive to use in the traditional boilers. The good tolerance for different fuel sources makes FBC well suited to also burn many types of biofuels. This is getting more and more important, as the pressure to reduce pollutions by replacing fossil fuels with carbon neutral alternatives is increasing globally.

Besides the currently existing FBC technologies, fluidization has also potential for even more advanced combustion designs. One prospective application of fluidization is the chemical looping combustion (CLC). In CLC there are two fluidized bed reactors operating in parallel. One of the reactors is a combustion reactor, where the fuel is burned within a bed of metal oxides. The oxides provide the oxygen required for the combustion. After the combustion, the reduced metals are transferred to the other fluidized bed, where they are re-oxidized with air. The CLC is an interesting concept, because it could provide the benefits of pure oxygen burning without the high cost of oxygen separation with current technologies. With pure oxygen, the flue gases contain mostly water vapour and $C0_2$ and it is much easier to capture the $C0_2$ compared to normal situation, where most of the flue gases consist of nitrogen. The $C0_2$ capturing is one of the promising approaches to reduce $C0_2$ emissions in the future.

There are many more applications for fluidization besides those few mentioned here. More comprehensive lists of applications can be found in the literature, for example in the book by Kunii and Levenspiel [2].

2.3 Particle classification

In section 2.1 a general description of fluidization phenomena was given. However, when a bed of solid particles is being fluidized, the exact behaviour will largely depend on the various properties of the particles in the bed. In order to predict the behaviour of the bed, it is thus essential to know what are the important properties and what kind of effect they have.

For flow dynamic considerations typically the most important properties to account for are the diameter, density and shapes of the particles. For completely spherical particles the diameter is a well-defined quantity. For irregular particles, however, the situation is not so clear. A common approach to describe a non-spherical particle is to define an equivalent spherical diameter:

$$d_{\mathrm{p},i} = \sqrt[3]{\frac{6\mathrm{V}_{\mathrm{p},i}}{\pi}}, \tag{1}$$

where $\mathrm{V}_{\mathrm{p},i}$ is the particle volume. [2] A sphere with the given equivalent diameter has the same volume as the particle being considered. For fully spherical particles their equivalent diameter is naturally the same as their real diameter.

After obtaining the equivalent diameter, the sphericity of a particle can be defined as the ratio of the particle's surface area $A_{\mathrm{sp},i}$ to the area of an equivalent volume sphere $A_{\mathrm{p},i}$:

$$\phi_{\mathrm{p},i} = \frac{A_{\mathrm{sp},i}}{A_{\mathrm{p},i}}. \tag{2}$$

The sphericity value has a range from one to zero, with one representing a perfect sphere.

A mixture of particles is usually described by using some form of mean values for the different properties of the single particles. The average density of n particles can be calculated as a simple weighted sum

$$\rho_{\mathrm{s}} = \frac{\sum_{i=1}^{n} \rho_{\mathrm{p},i} V_{\mathrm{p},i}}{\sum_{i=1}^{n} V_{\mathrm{p},i}}. \tag{3}$$

For calculating the average particle diameter there are several useful weighting schemes available. The different schemes can be conveniently presented by using unified notation

$$\mathrm{D}[k,l] = \frac{\sum_{i=1}^{n} d_{\mathrm{p},i}^{k}}{\sum_{i=1}^{n} d_{\mathrm{p},i}^{l}}. \tag{4}$$

The simplest weighting option is the standard arithmetic mean, also known as number-length mean:

$$\mathrm{D}[1,0] = \frac{\sum_{i=1}^{n} d_{\mathrm{p},i}}{n}. \tag{5}$$

While this mean is relatively general purpose, quite often it is necessary to use other mean values. For example, when considering catalytic reactions, which are proportional to the surface area of the particles, the number-surface mean

$$\mathrm{D}[2,0] = \sqrt{\frac{\sum_{i=1}^{n} d_{\mathrm{p},i}^{2}}{n}} \tag{6}$$

gives a more representative description of the mixture. Similarly, in some other situations where the volume of the particles is important the number-volume mean

$$\mathrm{D}[3,0] = \sqrt[3]{\frac{\sum_{i=1}^{n} d_{\mathrm{p},i}}{n}} \tag{7}$$

could be used. In fluidization research literature, it is common practice to use the Sauter Mean Diameter (SMD), defined as

$$\mathrm{D}[3,2] = \frac{\sum_{i=1}^{n} d_{\mathrm{p},i}^{3}}{\sum_{i=1}^{n} d_{\mathrm{p},i}^{2}}. \tag{8}$$

The SMD gives the diameter of a sphere with the equal volume to surface area ratio as in the mixture. The SMD is used in situations where the active surface area is important. It has been experimentally found out that the SMD is a good measure both for the fluid dynamic aspect of the particles and also for the possible chemical reactions occurring in the bed. [5] In this thesis the SMD is used by default when the diameter of solid particles is required, ie.

$$d_{\mathrm{s}} = \mathrm{D}[3,2].$$

Another quite often used mean value is the volume moment mean

$$\mathrm{D}[4,3] = \frac{\sum_{i=1}^{n} d_{\mathrm{p},i}^{4}}{\sum_{i=1}^{n} d_{\mathrm{p},i}^{3}}, \tag{9}$$

which places more emphasis on the volume of the particles.

The combination of the various properties of particles determines the behaviour of fluidized beds. For classification of bed materials, the Geldart [6] classification is widely recognized. In the Geldart classification the particles are classified in four different groups according to their density and diameter. The classification gives a practical tool to predict the behaviour of a fluidized bed, since each group has a clearly recognizable typical behaviour. Figure 3 presents the different groups in a diameter-density diagram. Originally the classification was made for relatively low velocity air fluidization at ambient conditions and various extensions can be found in literature.

The particles in group A have a small mean particle size and low density ($\rho \leq 1400\,\mathrm{kg/m^3}$). The particles in this class are easy to fluidize and the fluidization is smooth. The minimum bubbling velocity is clearly higher than the minimum fluidization velocity and the bed expands considerably in volume before the bubbling begins. There are many small bubbles, which split and re-coalesce very frequently and the bubbles have a maximum size. FCC catalyst is a typical example of a group A particle.

The particles in group B can be described as sand-like with a medium size and density ($1400\,\mathrm{kg/m^3} \leq \rho \leq 4000\,\mathrm{kg/m^3}$, $40\,\mathrm{\mu m} \leq d_\mathrm{p} \leq 500\,\mathrm{\mu m}$). The main difference compared to group A is that bubbles start to appear almost immediately after the minimum fluidization velocity. Also the bed expansion is smaller. The bubbles tend to get large without clear upper size limit and the bubbling is more vigorous. For both group A and group B particles the bubbles typically rise faster than the velocity of the interstitial gas.

The particles in group C consist of cohesive and very fine powders. These are very hard to fluidize as the particles tend to stick together. The resulting flow is typically very badly channelled or slug flow. Flour, starch, electrically charged or wet materials are typical examples of group C particles.

Finally group D consists of large and dense particles, such as gravel or coffee beans. Particles in this group are also quite difficult to fluidize and the behaviour of the fluidized beds is erratic. Large exploding bubbles and deep spouting beds are characteristic for particles in this size group. Also most bubbles in class D beds have slower velocity than the interstitial gas.

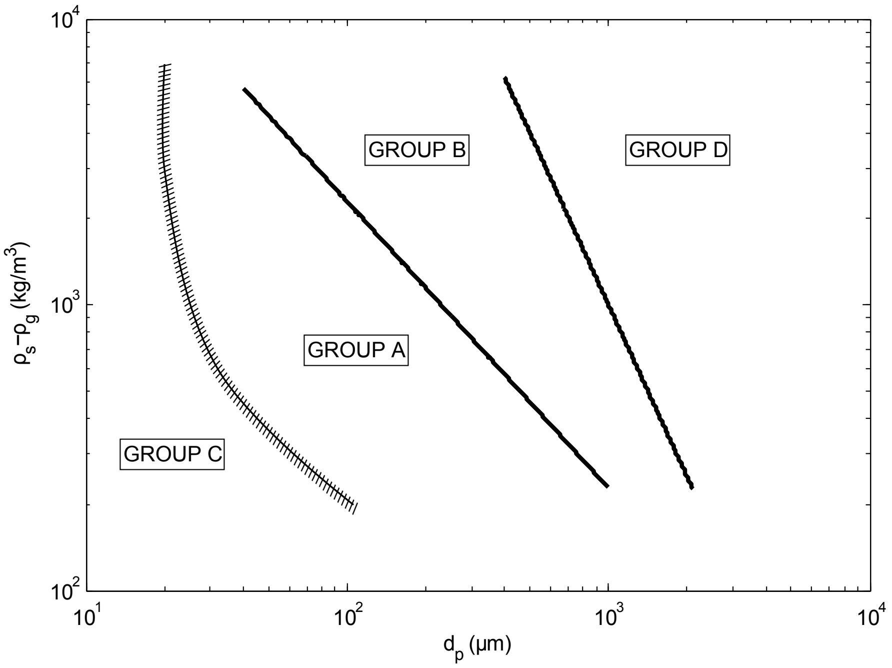

Figure 3: The Geldart classification of particles for air at ambient conditions. Figure is based on the work of Geldart. [6]

2.4 Minimum fluidization velocity and terminal velocity

One of the most important design parameters for a fluidized bed is the minimum fluidization velocity U_{mf}, since at that velocity the drag forces and the gravitational forces are in balance and the bed starts to exhibit fluidized behaviour. An often used empirical equation for U_{mf} can be written as

$$150\frac{(1-\varepsilon_{\mathrm{mf}})^2}{\varepsilon_{\mathrm{mf}}^3}\frac{\mu_{\mathrm{g}}U_{\mathrm{mf}}}{(\phi_{\mathrm{s}}d_{\mathrm{s}})^2}+1.75\frac{(1-\varepsilon_{\mathrm{mf}})}{\varepsilon_{\mathrm{mf}}^3}\frac{\rho_{\mathrm{g}}U_{\mathrm{mf}}^2}{\phi_{\mathrm{s}}d_{\mathrm{s}}}=(1-\varepsilon_{\mathrm{mf}})(\rho_{\mathrm{s}}-\rho_{\mathrm{g}})g, \tag{10}$$

where $\varepsilon_{\mathrm{mf}}$ is the voidage at minimum fluidization state, ρ_{s} and ρ_{g} are the densities of the solid and gas phases, g is the gravitational acceleration and μ_{g} is the fluid viscosity. This equation is known as Ergun equation and it is derived from extensive experimental observations and considered to be valid for many different fluidization cases. [7] To be able to use the equation, however, one has to know all the physical terms appearing in it. The density terms are usually known and the diameter and the sphericity of the particles should be calculated as presented in the previous section. The voidage term $\varepsilon_{\mathrm{mf}}$ is defined as the ratio of the total gas volume to the bed volume at the minimum fluidization state. It can be estimated from random packing data or obtained from the literature, but in order to get an accurate U_{mf} value, the voidage should be determined experimentally. Typically the voidage is around 0.4, but it varies depending on the material. [2]

It should be noted, that the value of U_{mf} obtained from the Ergun equation is the *superficial* minimum fluidization velocity. Superficial velocity is a velocity, which gives the correct volume flow rate in a situation, where the flow occupies the entire cross-sectional area. It is a useful measure, as it is independent of the geometry of the fluidized bed. As a consequence of the definition, in real fluidized bed systems, where the fluid is typically brought to the bed by small nozzles, at minimum fluidization the actual local flow velocities through the nozzles will be much higher than the U_{mf} value to provide the same overall flow rate.

If one wishes to simulate a BFB, it is very important to be able to predict the minimum fluidization velocity correctly. In a BFB the bed mass between the bubbles is usually roughly at minimum fluidization state and large amount of the fluidizing gas is used to keep it fluidized. Only the extra flow not required for this base fluidization creates the visible bubbles. This means that especially with low fluidization velocities the errors in the minimum fluidization velocity can have a substantial impact on the amount of gas available for the bubbles. If for example the simulation model gives a little bit higher minimum fluidization velocity than in reality, all the air flow could be used for fluidization with only few bubbles appearing and the simulation results would be likely wrong. Of course with high flow velocities this problem is not so serious as the relative amount of extra air increases.

When simulating or designing CFBs, another important quantity to estimate is the terminal velocity of the particles, ie. the velocity of a single particle in a free fall. In a CFB the particle concentration decreases along the height of the riser and near the top the interaction between the particles is quite low. Thus in order to keep the particles suspended and circulating, the flow velocity has to exceed the terminal velocity. The terminal velocity v_{t} can be estimated by considering the force balance of a particle in a free fall. For a single particle the balance between the drag and the gravitational forces can be expressed as

$$\frac{1}{2} C_{\mathrm{D}} \rho_{\mathrm{g}} A_{\mathrm{p}} v_{\mathrm{t}}^2 = (\rho_{\mathrm{s}} - \rho_{\mathrm{g}}) g V_{\mathrm{p}}, \tag{11}$$

where v_{t} is the terminal velocity, C_{D} the drag coefficient, A_{p} the cross-sectional area of a particle and V_{p} is the particle volume. The only complication of using this drag equation is the drag coefficient, which is generally speed dependent. Typically also the velocities in fluidized beds fall within the transition region between the low speed Stokesian drag and the high speed Newtonian drag. The Schiller-Naumann correlation is a well know empirical relation which can be used to estimate the drag coefficient in these cases. The correlation can be written as

$$C_{\mathrm{D}} = \begin{cases} \frac{24}{\mathrm{Re}_{\mathrm{t}}} \left[1 + 0.15 (\mathrm{Re}_{\mathrm{t}})^{0.687}\right] & \mathrm{Re}_{\mathrm{t}} \leq 1000 \\ 0.44 & \mathrm{Re}_{\mathrm{t}} > 1000, \end{cases} \tag{12}$$

where Re_{t} is the Reynold's number defined as

$$\mathrm{Re}_{\mathrm{t}} = \frac{\rho_{\mathrm{g}} \mathbf{v}_{\mathrm{t}} d_{\mathrm{p}}}{\mu_{\mathrm{g}}}. \tag{13}$$

2.5 Effect of the particle diameter and the size distribution

As it was stated in section 2.3, the average diameter of the particles is one of the most important factors affecting fluidized bed behaviour. For example, the diameter plays a big role in the Geldart classification of a fluidized bed, as can be seen in the figure 3. Also the important minimum fluidization and terminal velocities of the particles depend largely on the diameter and even relatively small changes in the particle sizes affect these velocities substantially. However, it is not enough to only consider the average sizes of the particles, since also the particle size distribution affects the bed behaviour. For example, in CFBs it is typical that the largest particles tend to remain near the bottom of the bed in dense suspension while the smaller particles flow more freely in the upper region. If one performs a simulation using only the average diameter for the whole bed, it can be hard to predict the proper solid distribution in the vertical direction.

The shape of the size distribution also affects the pressure drop and minimum fluidization behaviour of the bed. Figure 4 shows the pressure drop of the fluidizing gas as a function of velocity for two different particle size distributions. On the left image the particles have nearly uniform size and on the right the size distribution is wide. With uniform particles the pressure drop increases linearly with the velocity until the minimum fluidization velocity is reached. After that point the pressure drop decreases a bit and then stays almost constant in the bubbling region. On the other hand, with a wide size distribution the minimum fluidization velocity is not that well defined, since the smaller particles start to fluidize earlier than the larger particles and the bed can be in a partially fluidized state. In these cases U_{mf} can be defined as shown in figure 4. In addition to the influence on pressure drop, size distribution can also affect the voidage of the bed. The smaller particles can fit in the holes which are in between the bigger particles and that way reduce the empty space inside the bed. [2]

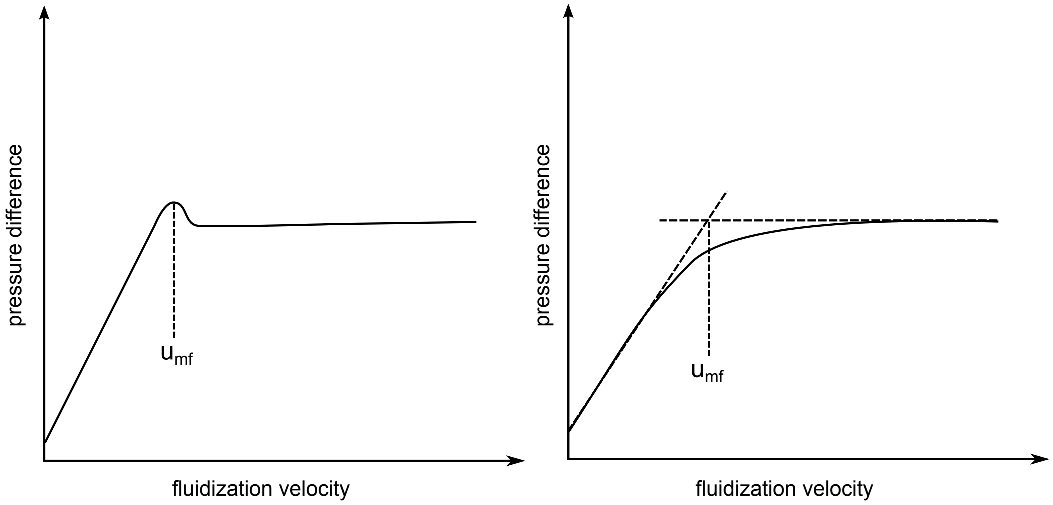

Figure 4: Pressure difference as a function of fluidization velocity for uniformly sized particles on the left and for wide size distribution on the right.

3 Multiphase Flow Simulation

3.1 Introduction

In this chapter some parts of the general theoretical background of multiphase flow simulation are presented. The multiphase flow is quite a general term and besides the gas-solid flows, also many other flow situations fall in the same category. Despite the similar overall setting, the exact models and computational techniques can differ quite a lot from case to case. The presentation here concerns mainly relatively dense gas-solid flows and all the models and approaches described in detail can be used for this kind of simulation. More general presentations of multiphase flows can be found in the literature, eg. in the books by Heng and Tu, Fan and Zhu and Kleinstreurer. [8–10]

In dense gas-solid flow simulation there are two basic approaches how to model the solid particles: the Eulerian-Eulerian approach and the Eulerian-Lagrangian approach. In the former of these two, the solid particles are considered to form a pseudo-fluid and they are modelled with equations similar to the classical Navier-Stokes flow equations. The Eulerian approach is currently commonly used in fluidized bed simulations and it is a relatively fast and usable approach. It has been implemented in various commercial codes and it is also the method of choice in this thesis. The drawback of the pseudo-fluid assumption is the requirement of complex constitutive relations, which are required to close the flow equations. A description of the constitutive relations, sub-models and assumption made in the Eulerian approach are given in the next section.

In the Eulerian-Lagrangian approach on the other hand, the particles are treated as separate, solid objects and their behaviour is modelled using Newtonian equations of motion. Depending on the implementation, the Eulerian-Lagrangian methods can give very accurate and detailed results in wide range of different settings. However, the computational costs of these approaches are often very demanding especially in case of dense suspensions and their application is mainly limited to smaller scale computations and research use. Brief survey of the various Eulerian-Lagrangian methods is given in section 3.4.

When considering particle size distributions, the Eulerian and Lagrangian approaches allow very different solution strategies. In the Lagrangian approaches the PSD can be usually included with relatively little extra coding and computational effort, since the properties of the each individual simulated particle, such as the diameter, can be typically chosen quite freely. On the other hand, in the basic Eulerian setting only the average particle properties are used and the inclusion of PSD is not as straightforward. Additional equations or sub-models have to be added to the simulation and depending on the exact approach chosen, the computational demands can be substantially increased. Details about the inclusion of PSD in Eulerian modelling are presented in the chapter 4.

There is also a hybrid modelling approach available, the so called dense DPM model, which combines aspects from both of the Eulerian and Lagrangian approaches. The dense DPM is a relatively new and promising approach, but currently it has not yet achieved wide spread adoption. The dense DPM method is described in section 3.5.

This thesis concentrates on time dependent flow simulation, which is currently the common approach with dense gas-solid flows. Time dependent flow simulation is computationally very demanding, as short time steps and fine computational meshes have to be used to get accurate results. [11] Also, with industrial processes the average behaviour of the flow is often the property that is interesting and to obtain good average values, long transient simulation periods are required. With single phase flows, the simulations are usually performed straight at steady state by using time averaged flow equations and suitable closure models. Steady state calculation could also speed up multiphase simulations by orders of magnitude, but unfortunately, the single phase closure models are not directly applicable. Closure models suitable for fluidized bed simulations are currently under active ongoing research. [12] Some aspects of time average simulation and the related field of turbulence modelling are discussed in the sections 3.6 and 3.7.

3.2 Eulerian-Eulerian approach

Currently the typical approach to model fluidized beds is the Eulerian-Eulerian model which is also known as the two fluid model (TFM). In this approach, both the gas phase and the granular solid phase are modelled as continuums and generalized Navier-Stokes equations are used for both phases. The continuum model for solids is based on the kinetic theory of granular flow (KTGF), which creates an analogy between the granular phase and the kinetic theory of dense gases. The full derivation of the two fluid equations and the KTGF is a rather long process and outside the scope of this thesis. Only the main concepts and equations are presented here, but interested reader can find the complete derivations in the literature. For the derivation of the KTGF one can consult [13–16] and the general discussion about the multiphase flow equations presented here is based on the work of Hiltunen et al. [17].

The equations in the following text do not include the energy equation and they are formulated for a single solid phase and particle size without considering mass sources or mass transfer between the phases. The limitation to single solid phase is mostly done for reasons of clarity, since the presented equations extend to multiple phases without much difficulty. Also, because the KTGF is a complex and semi-empirical model, there are a lot of choices for the various sub-models available in the literature and it's impractical to list them all. The relations presented here are some of the more common ones and implemented in commercial codes such as in FLUENT. [1]

The standard Navier-Stokes equations with a constitutive relation such as the Newtonian model are very general and sufficient to describe a large variety of single phase flows. For multiphase flows, however, the situation is not as clear and no standard set of both useful and widely applicable equations exist. We can derive quite general equations with only a small number of assumptions, but these equations will require complex constitutive relations. The modelling of these relations is a challenging task and due to the diverse nature of different multiphase flows the relations will be largely case dependent. The KTGF is a set of constitutive relations developed for dense gas-particle flows.

In the formal derivation of multiphase equations we typically first assume that inside each phase the normal continuity and momentum equations hold. At the boundaries of the different phases we can formulate boundary conditions which take into account the mass and momentum transfer and the possible surface tension between the phases. However, because we are considering situations where the phases are completely mixed together, the boundaries become very complex and some sort of averaging procedure is required to proceed. There are different approaches to do the averaging, such as time, volume or ensemble averaging and each of these approaches gives a different interpretation for the various interaction terms in the equations. Regardless of the chosen averaging scheme, however, in the end we should arrive to a set of similar equations. These equations give a general description of multiphase flow, but they still need closure models to apply to computations.

The averaging procedure introduces us the concept of volume fractions (VOF). A volume fraction is a scalar variable and its interpretation depends on the chosen averaging method. For example in the case of ensemble averaging a volume fraction of a phase can be considered to represent the probability of finding that phase in the given place at the given time. In this thesis the symbol α_{q} will be used to represent the volume fraction of phase q. With the volume fractions we can define the volume of a phase q as

$$\mathrm{V}_q = \int\limits_{\mathrm{V}(t)} \alpha_q \, d\mathrm{V} \tag{14}$$

and for n phases we have the constraint

$$\sum_{i=1}^{n} \alpha_i = 1. \tag{15}$$

In the KTGF it is assumed that the individual particles fluctuate randomly with a Gaussian normal distribution around their mean velocities. It is further assumed that the fluctuations are isotropic and a variable called granular temperature is defined as

$$\Theta_{\mathrm{s}} = \frac{1}{3}\langle c^2 \rangle, \tag{16}$$

where c is the fluctuation velocity of the solid particle and the brackets represent ensemble averaging. The particle phase viscosities and pressure are functions of this temperature and in addition to the equations of motion a transport equation is solved for the granular temperature. Unfortunately, the assumption of isotropy is valid only locally and thus when using the KTGF model in simulations a relatively fine computation mesh is needed to achieve good accuracy.

The multiphase flow equations obtained after the averaging procedure consist of a continuity and momentum equation for each phase. These equations look very similar to the normal N-S equations but there are some additional terms related to the interactions between the phases and there is also the solid pressure term for the particle phase. The continuity equation for a generic phase q and the momentum equations for gas g and solids s can be written as follows:

$$\frac{\partial}{\partial t}(\alpha_q \rho_q) + \nabla \cdot (\alpha_q \rho_q \mathbf{v}_q) = 0 \tag{17}$$

$$\frac{\partial}{\partial t}(\alpha_{\mathrm{g}} \rho_{\mathrm{g}} \mathbf{v}_{\mathrm{g}}) + \nabla \cdot (\alpha_{\mathrm{g}} \rho_{\mathrm{g}} \mathbf{v}_{\mathrm{g}} \mathbf{v}_{\mathrm{g}}) = -\alpha_{\mathrm{g}} \nabla p + \nabla \cdot \tau_{\mathrm{g}} + \alpha_{\mathrm{g}} \rho_{\mathrm{g}} \mathbf{g} + K_{\mathrm{gs}}(\mathbf{v}_{\mathrm{s}} - \mathbf{v}_{\mathrm{g}}) \tag{18}$$

$$\frac{\partial}{\partial t}(\alpha_{\mathrm{s}} \rho_{\mathrm{s}} \mathbf{v}_{\mathrm{s}}) + \nabla \cdot (\alpha_{\mathrm{s}} \rho_{\mathrm{s}} \mathbf{v}_{\mathrm{s}} \mathbf{v}_{\mathrm{s}}) = -\alpha_{\mathrm{s}} \nabla p - \nabla p_{\mathrm{s}} + \nabla \cdot \tau_{\mathrm{s}} + \alpha_{\mathrm{s}} \rho_{\mathrm{s}} \mathbf{g} + K_{\mathrm{gs}}(\mathbf{v}_{\mathrm{g}} - \mathbf{v}_{\mathrm{s}}) \tag{19}$$

where α_q is the volume fraction and $\mathbf{v}_{\mathrm{q}}$ the velocity of phase q, p the gas phase pressure, p_{s} the solid phase pressure, K_{gs} the inter-phase drag coefficient and ρ_{g} and ρ_{s} are the gas and solid densities. It is assumed that the stress tensor follows the Newtonian strain-rate relation:

$$\tau_q = \alpha_q \mu_q (\nabla \mathbf{v}_q + \nabla \mathbf{v}_q^T) + \alpha_q (\lambda_q - \frac{2}{3} \mu_q)(\nabla \cdot \mathbf{v}_q)\mathbf{I}, \tag{20}$$

where μ_q and λ_q are the shear and bulk viscosities and $\mathbf{I}$ the unit tensor.

The drag coefficient K_{gs} takes into account the interaction forces between the phases and it is a very important term in the equations. A more detailed discussion about the models used for this term is in section 3.3.

The solid pressure represents the additional normal stresses caused by the kinetics and collisions of the particles. Several different models for the pressure exist in the literature, eg. the models by Lun et al. [18] , Syamlal & O'Brien [19] and Ahmadi & Ma [20]:

$$p_{\mathrm{s}} = \alpha_{\mathrm{s}} \rho_{\mathrm{s}} \Theta_{\mathrm{s}} + 2\rho_{\mathrm{s}}(1 + e_{ss})\alpha_{\mathrm{s}}^2 g_{0,\mathrm{ss}} \Theta_{\mathrm{s}} \tag{21}$$

$$p_{\mathrm{s}} = 2\rho_{\mathrm{s}}(1 + e_{ss})\alpha_{\mathrm{s}}^2 g_{0,\mathrm{ss}} \Theta_{\mathrm{s}} \tag{22}$$

$$p_{\mathrm{s}} = \alpha_{\mathrm{s}} \rho_{\mathrm{s}} \Theta_{\mathrm{s}} \left[\left(1 + 4\alpha_{\mathrm{s}} g_{0,\mathrm{ss}}\right) + \frac{1}{2}(1 + e_{ss})(1 - e_{ss} + 2\mu_{\mathrm{fr}}) \right]. \tag{23}$$

In the equations above, e_{ss} is the restitution coefficient which describes the elasticity of the collisions between the particles. It is an empirical coefficient and its value should lie between one and zero. For fully elastic collisions $e_{\mathrm{ss}} = 1$ and for fully inelastic $e_{\mathrm{ss}} = 0$. For typical particles e_{ss} can be approximated to be quite close to one, eg. $e_{\mathrm{ss}} = 0.9$. The term μ_{fr} appearing at the end of equation (23) is the so called frictional viscosity term and its definition is given later in this section.

The term $g_{0,\mathrm{ss}}$ is the radial distribution function which is used in many relations to represent the amount of collisions between the particles. The basic kinetic theory assumes binary collisions and the $g_{0,\mathrm{ss}}$ is related to the probability of these collisions as a function of solid volume fraction. Again there are several alternatives for this function, but a commonly used definition by Ogawa et al. [21] is

$$g_{0,ss}(\alpha_\mathrm{s}) = \left[1 - \left(\frac{\alpha_\mathrm{s}}{\alpha_\mathrm{s,max}}\right)^{1/3}\right]^{-1}, \tag{24}$$

where $\alpha_{s,max}$ is the packing limit. The packing limit is an empirical coefficient which describes the maximal amount of packing possible and the radial function forces the volume fraction to be less than this. The radial functions typically have the following properties:

$$g_{0,ss}(0) = 1$$
$$\lim_{\alpha_\mathrm{s}\to\alpha_{s,max}} g_{0,ss}(\alpha_\mathrm{s}) = \infty.$$

The shear viscosity of the solids is considered to be a sum of collisional and kinetic terms and it can be written as

$$\mu_\mathrm{s} = \mu_\mathrm{s,col} + \mu_\mathrm{s,kin}. \tag{25}$$

For the both viscosity terms we have the models from Gidaspow [16]

$$\mu_\mathrm{s,col} = \frac{4}{5}\alpha_\mathrm{s}\rho_\mathrm{s}d_\mathrm{s}g_{0,\mathrm{ss}}(1+e_{ss})\left(\frac{\Theta_\mathrm{s}}{\pi}\right)^{\frac{1}{2}} \tag{26}$$

$$\mu_\mathrm{s,kin} = \frac{10\rho_\mathrm{s}d_\mathrm{s}\sqrt{\Theta_\mathrm{s}\pi}}{96\alpha_\mathrm{s}(3-e_{ss})g_{0,\mathrm{ss}}}\left[1+\frac{4}{5}(1+e_{ss})g_{0,\mathrm{ss}}\alpha_\mathrm{s}\right]^2, \tag{27}$$

where d_s is the particle diameter. Alternatively for the kinetic viscosity we have by Syamlal et al. [19]

$$\mu_\mathrm{s,kin} = \frac{\rho_\mathrm{s}d_\mathrm{s}\sqrt{\Theta_\mathrm{s}\pi}}{6(3-e_{ss})}\left[1+\frac{2}{5}(1+e_{ss})(3e_{ss}-1)g_{0,\mathrm{ss}}\alpha_\mathrm{s}\right]. \tag{28}$$

The granular bulk viscosity λ_s appearing in the solid phase strain tensor describes the resistance of the solid phase against volume changes. A model from Lun et al. [18] gives

$$\lambda_\mathrm{s} = \frac{4}{3}\alpha_\mathrm{s}\rho_\mathrm{s}d_\mathrm{s}g_{0,\mathrm{ss}}(1+e_{ss})\left(\frac{\Theta_\mathrm{s}}{\pi}\right)^{\frac{1}{2}}. \tag{29}$$

Finally the transport equation for the granular temperature can be written as

$$\frac{3}{2}\left[\frac{\partial}{\partial t}(\alpha_\mathrm{s}\rho_\mathrm{s}\Theta_\mathrm{s}) + \nabla\cdot(\alpha_\mathrm{s}\rho_\mathrm{s}\mathbf{v}_\mathrm{s}\Theta_\mathrm{s})\right] = (-p_\mathrm{s}\mathbf{I}+\boldsymbol{\tau}_\mathrm{s}):\nabla\mathbf{v}_\mathrm{s} + \nabla\cdot(k_{\Theta_\mathrm{s}}\nabla\Theta_\mathrm{s}) - \gamma_{\Theta_\mathrm{s}} + \phi_\mathrm{gs}. \tag{30}$$

For the diffusion coefficient we have again the models from Gidaspow [16]

$$k_{\Theta_\mathrm{s}} = \frac{150 d_\mathrm{s} \rho_\mathrm{s} \sqrt{\Theta_\mathrm{s}\pi}}{384(1+e_{ss}) g_{0,\mathrm{ss}}} \left[1 + \frac{6}{5}\alpha_\mathrm{s} g_{0,\mathrm{ss}}(1+e_{ss})\right]^2 + 2 d_\mathrm{s} \alpha_\mathrm{s}^2 \rho_\mathrm{s}(1+e_{ss}) g_{0,\mathrm{ss}} \sqrt{\Theta_\mathrm{s}/\pi} \quad (31)$$

and from Syamlal et al. [19]

$$k_{\Theta_\mathrm{s}} = \frac{15 d_\mathrm{s} \rho_\mathrm{s} \alpha_\mathrm{s} \sqrt{\Theta_\mathrm{s}\pi}}{4(41-33\eta)} \left[1 + \frac{12}{5}\eta^2(4\eta-3)\alpha_\mathrm{s} g_{0,\mathrm{ss}} + \frac{16}{15\pi}(41-33\eta)\eta\alpha_\mathrm{s} g_{0,\mathrm{ss}}\right], \quad (32)$$

where

$$\eta = \frac{1}{2}(1+e_{ss}). \quad (33)$$

The collisional dissipation of energy term $\gamma_{\Theta_\mathrm{s}}$ can be obtained from Lun et al. [18]

$$\gamma_{\Theta_\mathrm{s}} = \frac{12(1+e_\mathrm{ss}^2) g_{0,\mathrm{ss}}}{d_\mathrm{s}\sqrt{\pi}} \alpha_\mathrm{s}^2 \rho_\mathrm{s} \Theta_\mathrm{s}^{3/2} \quad (34)$$

and the energy exchange term ϕ_gs from Gidaspow [16]

$$\phi_\mathrm{gs} = -3K_\mathrm{gs}\Theta_\mathrm{s}. \quad (35)$$

When the volume fraction of the solids is nearing the packing limit, the assumptions made in the KTGF are no longer valid. In dense suspensions the particles are so close to each other that the interactions between them are better described as frictional as opposed to collisional. For this reason, usually some sort of a frictional model is employed in addition to the basic KTGF. Johnson and Jackson suggested a model where frictional terms are added to the kinetic theory solid phase viscosity and pressure terms. [22] From Johnson and Jackson the frictional pressure is expressed as

$$p_\mathrm{fr} = \begin{cases} 0 & \alpha_\mathrm{s} < \alpha_\mathrm{s,min} \\ \mathrm{Fr}\frac{(\alpha_\mathrm{s}-\alpha_\mathrm{s,min})^n}{(\alpha_\mathrm{s,max}-\alpha_\mathrm{s})^p} & \alpha_\mathrm{s} \geq \alpha_\mathrm{s,min} \end{cases} \quad (36)$$

and the frictional viscosity as

$$\mu_\mathrm{s,fr} = p_\mathrm{fr}\sin\phi_f. \quad (37)$$

The terms Fr, n and p are empirical constants and the parameter $\alpha_\mathrm{s,min}$ defines the volume fraction after which the frictional model gets activated. The constant ϕ_f is called the angle of internal friction and it relates the normal stresses to the frictional stresses. For the frictional viscosity we have also the model of Schaeffer [23]

$$\mu_\mathrm{s,fr} = \frac{p_\mathrm{fr}\sin\phi}{2\sqrt{I_\mathrm{2D}}}, \quad (38)$$

where I_2D is the second invariant of the deviatoric strain tensor defined as

$$I_\mathrm{2D} = \frac{1}{6}\left((D_{11}-D_{22})^2 + (D_{22}-D_{33})^2 + (D_{33}-D_{11})^2\right) + D_{12}^2 + D_{23}^2 + D_{31}^2, \quad (39)$$

where

$$\mathbf{D} = \frac{1}{2}(\nabla \mathbf{v}_\mathrm{s} + \nabla \mathbf{v}_\mathrm{s}^T). \tag{40}$$

There are also other formulations of the frictional effects available. For example in the model of Syamlal et al. [19], instead of summation the shear stress switches to purely frictional after the volume fraction $\alpha_{\mathrm{s,min}}$ is exceeded.

3.3 Gas-solid drag models

Finding an accurate model for the interphase drag coefficient K_{gs} appearing in the momentum equations is a crucial task in fluidization modelling. Typically the drag term is the largest term in the equations of the vertical direction. It roughly balances the pressure gradient in the gas phase equations and the gravitational term in the solid phase equations. Because the drag term dictates when the gravitational and drag forces are in balance in fully developed flow conditions, it effectively determines the vertical solid distribution. Correct prediction of this distribution is a key requirement in CFB modelling. Also in dense suspensions, such as in bubbling beds or in the dense bottom area of a CFB riser, the drag term determines the minimum fluidization velocity. Even relatively small errors in this velocity can lead to unrealistic behaviour of the overall flow as described in section 2.4.

Perhaps the most widely used drag model is the approach proposed by Gidaspow [16], which is a combination of Ergun [7] and Wen & Yu [24] drag models. Both models are based on empirical findings. The Ergun model is directly derived from the Ergun equation (10) and it is considered to be valid especially in dense suspensions. On the other hand, the Wen & Yu model is based on a wide range of observations and it is considered to have a better accuracy in dilute flows. Thus a combination of these two models should give us an accurate description in both dense and dilute situations.

For the Erqun model the drag term is

$$K_{\mathrm{gs}} = K_{\mathrm{E}} = 150\frac{\alpha_\mathrm{s}(1-\alpha_\mathrm{g})\mu_\mathrm{g}}{\alpha_\mathrm{g} d_\mathrm{s}^2} + 1.75\frac{\rho_\mathrm{g}\alpha_\mathrm{s}\left|\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}\right|}{d_\mathrm{s}} \tag{41}$$

and for Wen & Yu it is

$$K_{\mathrm{gs}} = K_{\mathrm{WY}} = \frac{3}{4}C_\mathrm{D}\frac{\alpha_\mathrm{s}\alpha_\mathrm{g}\rho_\mathrm{g}\left|\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}\right|}{d_\mathrm{s}}\alpha_\mathrm{g}^{-2.65}, \tag{42}$$

where the coefficient C_d comes from the Schiller-Naumann correlation, which is written here again for convenience:

$$C_\mathrm{D} = \begin{cases} \frac{24}{\mathrm{Re_s}}\left[1 + 0.15(\mathrm{Re_s})^{0.687}\right] & \mathrm{Re_s} \le 1000 \\ 0.44 & \mathrm{Re_s} > 1000. \end{cases} \tag{43}$$

In this context the Reynold's number appearing in the correlation can be written as

$$\mathrm{Re_s} = \frac{\alpha_{\mathrm{g}}\rho_{\mathrm{g}}\left|\mathbf{v}_{\mathrm{g}} - \mathbf{v}_{\mathrm{s}}\right| d_{\mathrm{s}}}{\mu_{\mathrm{g}}}. \tag{44}$$

In the original work from Gidaspow the two models are combined in a discontinuous fashion:

$$K_{\mathrm{gs}} = \begin{cases} K_{\mathrm{WY}} & \alpha_{\mathrm{s}} < 0.2 \\ K_{\mathrm{E}} & \alpha_{\mathrm{s}} \geq 0.2. \end{cases} \tag{45}$$

This is still a common approach, but because of the unphysical discontinuity also other, smoother transitions have been suggested. One could use, for example, linear interpolation between the two models when the solid volume fraction is near to the transition point. In a study by Leboreiro et al. [25] it was found out that the use of smooth switching has a noticeable effect on the results. It was also noted that the point of switching has a clear impact on the results and they suggested to use a higher volume fraction for transition, with $\alpha_{\mathrm{s}} = 0.46$ being an appropriate value.

3.4 Eulerian-Lagrangian approach

Besides the Eulerian-Eulerian approach, fluid-particle flows can also be modelled with the so called Eulerian-Lagrangian methods. In these approaches the fluid phase is still modelled with Navier-Stokes equations, but the particle phase is modelled using a large amount of individual particles obeying the Newtonian equations of motion. There are many methods, with differing degrees of complexity, which can all be described as Eulerian-Lagrangian. The main differences between these lie in the way the particle-particle collisions and the interactions between the phases are modelled. Also in different approaches a numerical particle may act as a single physical particle or a representative for a larger collection of particles. In this section some of the common Eulerian-Lagrangian methods of modelling are presented.

In the simple situations where the flow is very dilute and the particles have only negligible influence on the main flow, the Eulerian-Lagrangian approach is straightforward to apply. In these cases we can separately solve the N-S equations for the fluid phase, and based on this solution calculate the particle trajectories. The simplest approximation is to use the same velocity field for the particles, so that they simply follow the flow. In a little bit more sophisticated approach one could use separate velocity fields for the two phases, but only assume a one-way coupling. This means that the trajectories of the particles are affected by drag forces and turbulent effects, but the fluid phase doesn't feel the particles' presence. Simple models like these are fast to compute and they can be suitable in eg. modelling small particle dispersion in the atmosphere. In the case of dense flows like in fluidization, however, the particle phase occupies a substantial part of the total volume and mass and the particle-particle and particle-fluid interactions can't be neglected. For these reasons, we have to use more complex models such as the soft- or the hard-sphere approaches.

A fairly recent review of the soft- and hard-sphere approaches, also known as Dense Particle Models (DPM), is given by Deen et al. [26]. The soft-sphere approach is the more complex of the two models. In that approach the particles are assumed to be elastic spheres and their positions are calculated using a fixed time step. Typically this time step is a small fraction of the main time step used in the flow field calculation. Because the particles are elastic, they can compress slightly in the collisions and the contact forces between them have to be calculated using a some sort of a contact model. Typically relatively simple models using spring and damping constants are used, but also more complex models taking account eg. the Van der Waals forces have been applied. The interaction with the fluid phase is modelled using a suitable drag model, such as the Gidaspow model presented in the previous section. A simplified diagram of the overall solution process is presented in figure 5.

With a proper contact scheme and a drag model, the soft-sphere approach can give us a very accurate description of the particle phase. Unfortunately, this accuracy comes with a very high computational cost. If the collisions between the particles

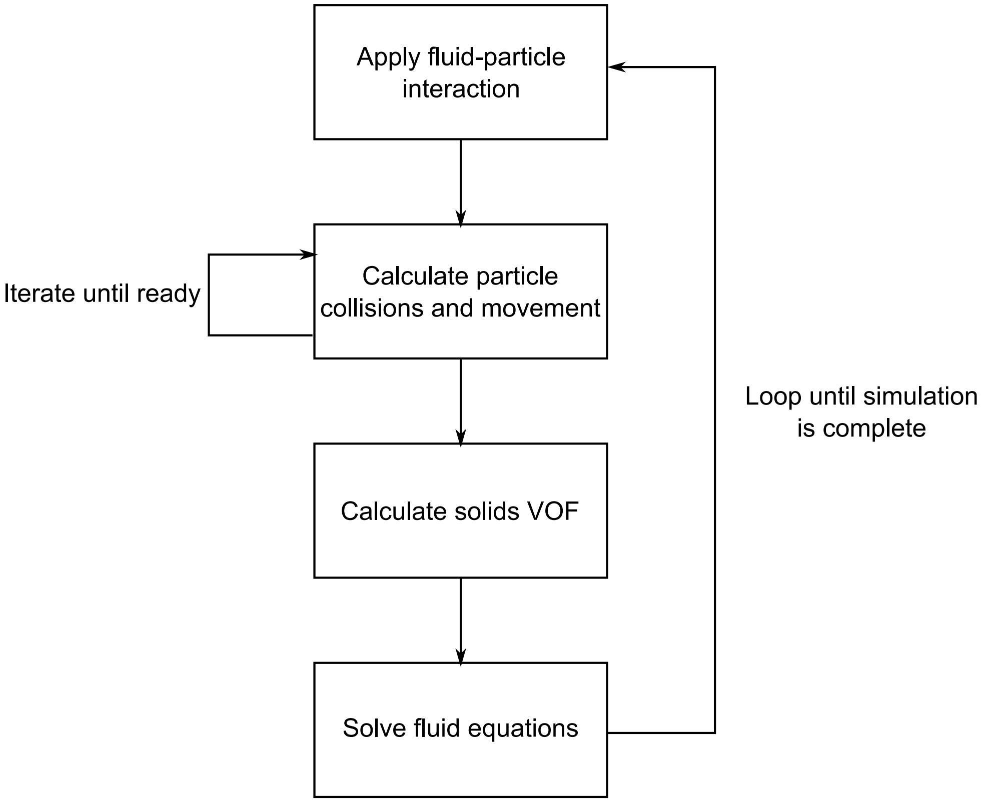
Apply fluid-particle interaction
Iterate until ready
Calculate particle collisions and movement
Loop until simulation is complete
Calculate solids VOF
Solve fluid equations

Figure 5: A simplified flow diagram of a generic dicrete particle model. Adapted from an image presented by Deen et al. [26].

are stiff, as they usually are, a very small time step has to be used in order to keep the particle simulation stable. In practice the amount of simulated particles is limited to perhaps a few millions and for any large scale simulation this is too little by many orders of magnitude.

The hard-sphere approach tries to ease the computational costs by assuming that the particles are rigid spheres. Only binary collisions are treated and they are considered to be instantaneous. With this approach it is possible to use an event based time scheme, where only the collisions are tracked. This can be done, for example, by finding all the possible colliding pairs within each main time step and assuming that the particles are in free flight between the collisions. The collisions can then be processed in the order they occur.

If the flow is dilute and there are only few collisions in each time step, the hard-sphere approach can be much faster than the soft-sphere approach. On the other hand, in dense flows the collision calculations can take a long time and the accuracy of the hard-sphere approach can suffer. If a restitution coefficient is used in collisions to allow kinetic energy dissipation, simple hard-sphere approaches are also susceptible to the so called inelastic collapse. This is a situation where closely packed particles go through a very large number of collisions in a small time frame and thus lose their kinetic energy.

Even with the much faster hard-sphere approach the amount of simulated particles is usually still too low for full scale circulated bed simulations. Despite this, the soft- and hard-sphere approaches can be considered very useful, because they are able to give us valuable insight when developing better closure models for other solution methods. It is also possible to extend these methods to larger scales by letting each simulated particle represent a group of similar real particles.

3.5 The dense DPM approach

Besides the pure Eulerian-Lagrangian methods it is also possible to use a hybrid approach, where the troublesome particle-particle collisions are modelled only statistically in the same fashion as in the KTGF. This kind of approach is known as the dense DPM (Discrete Particle Model) method, or alternatively as CPFD (Computational Particle Fluid Dynamic) method or MP-PIC (MultiPhase Particle-In-Cell) method depending on the author. The method was first presented for one dimensional case by Andrews and O'Rourke [27] and it has been subsequently extended to two- and three dimensional cases by Snider and O'Rourke [28] and Snider [29]. In a recent paper by Snider [30], the dense DPM method was successfully applied to several different 3D test cases which would be hard to simulate with the Eulerian-Eulerian approach.

In the dense DPM method the particle phase is simulated by tracking a large number of numerical parcels, which each represent a group of particles with similar properties. However, unlike in traditional Lagrangian methods, one also calculates

Eulerian velocity and volume fraction fields for the particles. The Eulerian fields are computed from the discrete representation by using suitable, conservative interpolation operators. The coupling between the representations is two-way and information can be also passed from the Eulerian representation to the discrete representation using the interpolation operators. However, the particle positions are modelled with Newtonian equations of motion and no momentum equation for particles is solved. The advantage of the Eulerian fields is the fact that the collisional forces and interaction with the fluid phase can be done through the Eulerian fields. This kind of procedure allows to simulate the particle-particle collisions stably without the need of very small time-steps and thus the most time consuming part of DPM simulations is avoided. This easily makes up for the additional cost of using two kinds of representations.

The dense DPM approach appears to be very interesting, since it has the benefits of the Lagrangian methods, but it should still be relatively fast to use. Like in other Lagrangian methods, it is very easy to model polydisperse flows with arbitrary size distributions. One can freely alter the diameters and amounts of the simulated particles and thus match any kind of real distribution. One could also give different densities or chemical properties to different particles and by that way easily model even very complex mixtures. In addition, since there is no need to solve the momentum equation for the particles, there are no issues with numerical diffusion normally present due to the discretization of the advection operator. Thus all in all it could be argued, that since we are indeed trying to simulate particulate behaviour, it would make more sense to really use particles in the numerical models.

Because of the great potential of the dense DPM approach, it was a good candidate to be chosen as the primary simulation model, since it could have easily solved the entire PSD problem. Unfortunately, after some initial research it was found out, that at least the existing implementation in FLUENT is currently severely limited even in the newest solver version (v.14) and is unsuitable for fluidization modelling. The implementation seems to be targeted at more dilute flow calculations, as the dense flow regions appeared to cause great deal of problems and the simulations gave very unphysical results. Also, the implementation does not strictly follow the models presented in the literature, since for example, the solids volume fraction is not directly interpolated from the particle positions. Because of these reasons, the dense DPM model had to be left out of consideration. Extending the FLUENT with own implementation would have been difficult to do and most likely the performance wouldn't have been satisfactory. Also writing an own solver or using some other solver was out of the scope of this thesis.

Despite the problems encountered with the dense DPM implementation in FLUENT, the method in itself still looks promising. Better implementations and more thorough validations of the method will most likely appear in the future and at some point the dense DPM may well become the method of choice with fluidization simulations. One should take into account, however, that since the continuous phase is simulated in the same way as in the normal Eulerian-Eulerian approach, the same limitations regarding for example turbulence and computational mesh size apply. Also, in order to get accurate results and good interpolation behaviour, there has to be enough simulated parcels in every computational cell. With large geometries or fine meshes this can cause considerable workloads even with the dense DPM method.

3.6 Turbulence modelling

Even with today's fast computers, in most practical flow computations the turbulent flow scales can't be completely resolved and one has to use some form of turbulence modelling in the simulations. Even though the turbulence phenomena are still not very well understood, with single phase flows there are several relatively good turbulence models available and turbulence modelling is routinely performed. Unfortunately, for gas-particle flows no generally valid turbulence models exist and the two-phase turbulence phenomena are even less understood than in the case of single phase flows. There are some turbulence models available, but they are mostly for dilute flows and even then their validity is questionable. Typically these models are derived from the single phase models, like from the $k-\epsilon$-model.

Gas-particle turbulence modelling is difficult, because the gas and solid phase velocity fluctuations are strongly linked together. Even relatively small amounts of particles start to affect the turbulent behaviour of the gas phase and in dense regions the turbulent fluctuations are very different from those appearing in single-phase flows. Proper turbulence modelling would require good knowledge of the particle-particle and particle-gas interactions in these dense regions, but unfortunately they are not very easy to measure in practice. Another complicating thing is the fact that in CFBs the turbulent fluctuations in dilute regions are also typically very anisotropic, with velocity fluctuations being substantially larger in the vertical direction than in the lateral directions. [31] This causes problems with simple turbulence models which assume that the turbulence is isotropic. Fortunately though, in the dense bottom part the fluctuations are typically more isotropic.

Despite the complications with the two-phase turbulence, new turbulence models are actively researched and appearing in the literature. For example, there is a sequence of improving models by Zhou et al. [31–33], which are similar to the single phase Reynolds stress model. Unfortunately, these models often tend to be quite complicated and they may require many empirically determined constants. For this reason, in practice the fluidized beds are still usually simulated by using some simple turbulence model for the local scale turbulence. Even though the models are not perfect, in dilute regions the results can be better with the models than without them and often the turbulence models also help to make the computations more stable. In the dense regions, where the models are not physically realistic, the other terms in the momentum equations, such as drag and gravitation, are much bigger and the errors in the turbulence model are not that significant for the end result.

3.7 Time averaged modelling

As mentioned in the introduction of this chapter, fluidized bed simulations are currently usually run as time dependent, which creates serious performance problems when the calculation of average flow fields for large industrial geometries is required. It would be much faster if the simulations were computed straight in steady state using suitable closure models like it is routinely done with single phase flows. Unfortunately, both the amount of closures needed and the impact of the closures tends to be much higher with multiphase flows than with single phase flows and this makes it difficult to develop robust closure models. Still, due to the potentially great performance gains, steady state models are actively studied and closure models are being developed. A recent example of this research is the time-averaged model developed by Taivassalo et al. [12], which is briefly presented here.

In the similar way as with the single phase flows, the time-averaged multiphase flow equations can be derived using the Reynold's averaging procedure. The Reynold's averaging is a very well known approach and is presented in detail in many general CFD books. [34, 35] In Reynold's averaging the instantaneous flow variables ϕ are split into steady and fluctuating parts:

$$\phi = \overline{\phi} + \phi', \tag{46}$$

where $\overline{\phi}$ represents the average value and ϕ' is the fluctuating part. The average value over some time interval T is defined as

$$\overline{\phi} = \frac{1}{T}\int_T \phi \, \mathrm{d}t \tag{47}$$

and for the fluctuating part the average vanishes

$$\overline{\phi'} = 0. \tag{48}$$

The single phase Reynold's averaged Navier-Stokes (RANS) equations can be obtained by taking the Reynold's average over the flow equations. After simplifications, the obtained equations are very similar to the normal N-S equations, but there are also additional stress terms, known as Reynold's stresses, which have to modelled using a suitable turbulence model.

With multiphase flows, or alternatively with single phase flows with a variable density, it is also useful to consider mass-weighted averaging, which is known as Favre averaging. Similarly as with the Reynold's decomposition, the instantaneous variables can be split in two parts

$$\phi = \langle\phi\rangle + \phi'', \tag{49}$$

where the Favre average is defined as (constant density, but variable VOF):

$$\langle \phi \rangle = \frac{\overline{\alpha_q \phi}}{\overline{\alpha_q}}. \tag{50}$$

Again, for the fluctuating part it holds

$$\langle \phi'' \rangle = 0. \tag{51}$$

Following the notations by Taivassalo et al. [12], the basic multiphase flow equations for a phase q can be written as

$$\frac{\partial \alpha_q \rho_{qm}}{\partial t} + \frac{\partial \alpha_q \rho_{qm} u_{q,k}}{\partial x_k} = 0 \tag{52}$$

$$\begin{aligned}\frac{\partial \alpha_q \rho_{qm} u_{q,i}}{\partial t} + \frac{\partial \alpha_q \rho_{qm} u_{q,k} u_{q,i}}{\partial x_k} =& -\alpha_q \frac{\partial p}{\partial x_i} + \frac{\partial \alpha_q \tau_{q,ik}}{\partial x_k} + \frac{\partial \alpha_q \tau^M_{q,ik}}{\partial x_k} - \frac{\partial p_q}{\partial x_i}\delta_{qs} \\ &+ \alpha_q \rho_{qm} g_i + (-1)^{(\delta_{qs}+1)} K_{\text{gs}}(u_{g,i} - u_{s,i})\end{aligned} \tag{53}$$

where δ_{qs} is the Kronecker delta and g and s refer to the gas and solid phases. These are the same equations as the equations (17)-(19), but here they are written more compactly with index notation and the local scale turbulent stresses τ^M are already included in the formulation.

Now when the Favre averaging is applied on the velocities and time averaging on the pressure and volume fractions, according to Taivassalo et al. the following equations are obtained:

$$\frac{\partial \overline{\alpha_q} \rho_{qm}}{\partial t} + \frac{\partial \overline{\alpha_q} \rho_{qm} U_{q,k}}{\partial x_k} = 0 \tag{54}$$

$$\begin{aligned}\frac{\partial \overline{\alpha_q} \rho_{qm} U_{q,i}}{\partial t} + \frac{\partial \overline{\alpha_q} \rho_{qm} U_{q,k} U_{q,i}}{\partial x_k} =& \overline{\alpha_q} \rho_{qm} g_i - \overline{\alpha_q} \frac{\partial \overline{p}}{\partial x_i} - \overline{\alpha'_q \frac{\partial p'}{\partial x_i}} + \frac{\partial \overline{\alpha_q \tau_{q,ik}}}{\partial x_k} \\ &+ \frac{\partial \overline{\alpha_q \tau^M_{q,ik}}}{\partial x_k} + (-1)^{(\delta_{qs}+1)} \overline{K_{gs}(u_{gi} - u_{si})} \\ &- \frac{\partial \overline{p_q}}{\partial x_i}\delta_{qs} - \frac{\partial \rho_{qm} \overline{\alpha_q u''_{q,k} u''_{q,i}}}{\partial x_k},\end{aligned} \tag{55}$$

where $U_{q,i} \equiv \langle u_{q,i} \rangle$.

As it can be seen from the averaged equations, several new terms have appeared due to the averaging process and for most of them a suitable closure model has to be utilized. Developing and validating all of these closures is a considerable task, since in addition to the terms shown here, the turbulence model for the Reynold's stress term introduces even more unclosed terms. Nevertheless, Taivassalo et al. managed to develop the closures based on data from a large number of transient simulations. The obtained results were in good agreement with the transient simulations and thus the feasibility of the method was proven. Of course, as the approach is still very new, further validation is required. However, if the time averaged methods can be shown to scale and work with different kinds of geometries, they would represent a large advancement in the practicality of CFD modelling for fluidized beds.

4 Modelling Particle Size Distribution

4.1 The population balance equation

In many engineering fields it is necessary to work with systems of particles or with other large collections of similar items. For example, besides in gas-solid flows, these kinds of systems are often encountered in pharmaceutical manufacture, when dealing with crystallisation or when studying the growth of cell populations. While the specifics of each system are different, what is common to most of them is the fact that the systems under study have a distribution of properties, which can change in time and perhaps also in space. A general theory describing the evolution of these kinds of systems is known as the population balance approach and the equations coming from this theory are called population balance equations (PBE). A thorough and modern treatment of the population balance theory can be found in the book by Ramkrishna [36] and this section is based on that presentation.

In the population balance approach, it is assumed that the particles are dispersed in an environmental phase, known as the continuous phase and that the particles can interact between themselves and with the continuous phase. The state of each particle is represented with two sets of coordinates: the internal and external coordinates. The external coordinates represent the spatial location of the particle and their value is determined by the actual physical motion. The internal coordinates represent the intrinsic properties of the particle, such as size, material, composition or any other interesting property and the movement in these coordinates depends on the application. Both continuous and discrete coordinate values can be used and they can also be mixed together freely. Together the internal and external coordinates form the state space of the particles.

In population balance theory, it is assumed, that there exists a number density function f, for example the particle size distribution, which gives the density of particles at every point in the particle state space and which can be integrated to obtain the total amount of particles in a given domain. The density function is the primary quantity to be determined and it is obtained by solving the corresponding population balance equation.

As its name states, the PBE is a balance equation for the population of the particles and it can be derived in the same way, as many other balance or continuity equations in the continuum mechanics are derived. In a non-rigorous fashion the derivation for these kinds of equations goes like this: first assume that there is an arbitrary (and time dependent) control volume Ω in the state space of the particles following the particle motion. Now by common reason, if we assume that there are no births or deaths of particles (not a necessary assumption), the total amount of particles within the control volume should stay constant in time. This can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}N = 0, \tag{56}$$

where N is the total number of particles, or by using the density function f as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega} f \,\mathrm{d}\Omega = 0. \tag{57}$$

By assuming that the density function f is sufficiently smooth, the time derivative can be moved inside the integral by applying the Reynold's transport theorem (see eq. [37]):

$$\int_{\Omega} \frac{\partial f}{\partial t} + \nabla \cdot (\mathbf{v} f) \,\mathrm{d}\Omega = 0, \tag{58}$$

where $\mathbf{v}$ is the particle velocity. Since the domain Ω was arbitrary, it can be concluded that the expression under the integral has to be zero, ie.

$$\frac{\partial f}{\partial t} + \nabla \cdot (\mathbf{v} f) = 0, \tag{59}$$

and thus we have derived a balance equation in differential form. More rigorous proofs, with various different assumptions taking account eg. source terms are presented by Ramkrishna [36].

A suitable form of the PBE for CFD applications can be expressed as

$$\frac{\partial f(\mathbf{x},\xi,t)}{\partial t} + \nabla \cdot (\mathbf{V}(\mathbf{x},\xi,t) f(\mathbf{x},\xi,t)) = S(\mathbf{x},\xi,t), \tag{60}$$

where f is the density function, $\mathbf{V}$ is the velocity vector in the external, ie. physical, space and S represents source terms. [8] The argument $\mathbf{x}$ refers to the physical coordinates, ξ to the internal coordinates and t is the time. The ξ can be either a vector or a scalar, depending on how many particle properties are deemed important. In this thesis, the only interesting internal coordinate is the diameter of the particles and therefore ξ is assumed to be scalar.

The source terms S depend on the actual physics of the particulate system under consideration and for example, in fluidized beds they could include terms describing particle aggregation and breakage. The source terms often contain integral expressions and they have to be closed by using suitable models to account for the physical processes in the system. For example, a formulation for a source term, which takes aggregation and breakage into account can be given as

$$\begin{aligned} S(\mathbf{x},\xi,t) = {} & \frac{1}{2}\int_0^{\xi} a(\xi-\xi',\xi') f(\xi-\xi',t) f(\xi',t) \,\mathrm{d}\xi' \\ & - f(\xi,t)\int_0^{\infty} a(\xi,\xi') f(\xi',t) \,\mathrm{d}\xi' \\ & + \int_{\xi}^{\infty} \gamma(\xi') b(\xi') p(\xi|\xi') f(\xi',t) \,\mathrm{d}\xi' - b(\xi) f(\xi,t), \end{aligned} \tag{61}$$

where the first two terms represent birth and death by aggregation and the last two terms represent birth and death by breakage. [8] The $a(\xi,\xi')$ is the aggregation rate between particles of size ξ and ξ', $b(\xi)$ is breakage rate for particles of size ξ, $\gamma(\xi')$

gives the amount of fragments in a breakage and $p(\xi|\xi')$ is the probability density function for the fragments of a breakage. The treatment of source terms is an integral part of the population balance theory and the choice of closure models can significantly affect the obtained results. However, in this thesis it is assumed that the particles stay completely intact and thus the source terms are omitted.

Since PBEs are partial differential equations and they possibly also contain integral terms, they are usually solved by numerical methods. There are many different solution strategies available, such as Monte Carlo methods, class methods (CM) and methods of moments (MOM). In the Monte Carlo methods, the solution of the PBE is based on statistical ensemble approaches. Generally speaking the Monte Carlo methods are flexible and it is possible to get very accurate solutions with them. Unfortunately though, they tend to be very computationally demanding and unsuitable in the context of CFD, where the basic flow simulation takes up most of the resources. For this reason, in most practical CFD settings some variant of either the CM or MOM approaches are used. [8]

In this chapter some popular CM and MOM approaches are presented and their strengths and weaknesses are studied. Although the theoretical starting points for the CM and MOM approaches are different, in the end their implementations in the CFD context are relatively similar. In both approaches the discretization of the PBE leads to few scalar transport equations, which have to be solved together with the normal flow equations (17)-(19). These transport equations can be implemented as additional continuity equations and depending on the approach, some additional momentum equations may have to be solved.

4.2 The class methods

The class methods offer what is maybe the most straightforward approach to solve the PBE and they are thus very popular solution methods and available in many solver codes. [8] In the CM the particle size distribution is discretized into a number of discrete size classes and for each size class a scalar PBE for the number density of particles in that class is solved. The PSD is thus expressed as

$$f(\mathbf{x},L,t) = \sum_{j=1}^{N} N_j(\mathbf{x},t)\delta[L-\{L\}_j], \tag{62}$$

where N_j is the number density for the jth size class, δ refers to Dirac delta function, L is the particle length and $\{L\}_j$ is the reference length for the jth size class. The discretization is determined before the simulations are started and initially, the particle density in each class is calculated from the true, continuous size distribution. Figure 6 visualizes this discretization process.

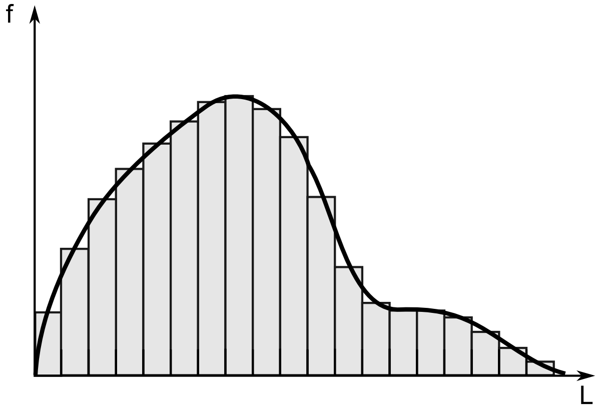

Figure 6: A particle size distribution and the corresponding discretization into different size classes.

Several alternative formulations of CM have been developed for various different applications and by different authors. One of the approaches is the multiple size group (MUSIG) model, first introduced by Lo, which is a popular model in gas-liquid flow simulations. [38] Currently, there are two different types of MUSIG models available: homogeneous and inhomogeneous. In the homogeneous model it is assumed, that all the particles share a same velocity field. For each particle size class, a continuity equation is added to the standard set of flow equations (17)-(19) and the source terms of these equations take into account the inter-class mass transfer mechanisms. Because the particle velocity field is shared, it is enough to solve only one momentum equation (19) for the whole solid phase, which is convenient, since it spares computation time.

The homogeneous MUSIG model has been successfully applied to some gas-liquid flows, although typically quite a large number of size classes are required for accurate results. Unfortunately, for gas-solid flows in fluidized beds, the assumption of a size independent velocity field is not physically valid. Especially in CFBs, where the particle size affects greatly the flow behaviour, there is considerable velocity slip between the differently sized particles. With a homogeneous velocity field many of the physically important features can therefore be lost. To remedy this problem, the inhomogeneous MUSIG model was developed by Krepper et al. [39] In the inhomogeneous MUSIG the particle phase is further divided into several velocity groups, which contain one or more size groups. This division is illustrated in figure 7.

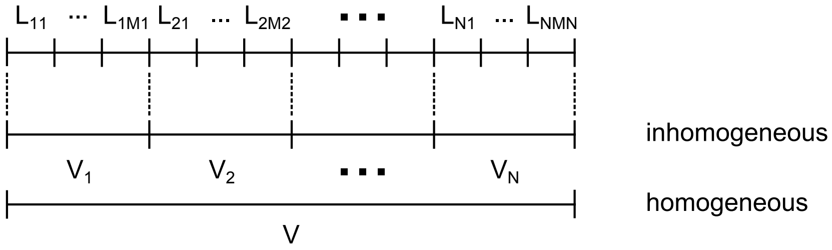

Figure 7: In the inhomogeneous MUSIG the particle size distribution is divided into N velocity groups, which in turn contain M size groups. In homogeneous MUSIG a single velocity field is used for all the particle sizes.

For each size group a continuity equation is solved, like in the homogeneous case, but now the velocity in each of these equations depends on which velocity group the size group belongs into. The velocities for each velocity group are obtained from momentum equations and for each group a separate equation has to be solved. The idea is, that the particle phase is modelled as if it consists of several different phases with different diameters. It is possible to do many kinds of combinations with the velocity and size groups, with one extreme being the homogeneous case and the other extreme being separate velocity fields for each size class, in which case each of the size classes acts like a separate Eulerian phase in the flow equations.

From theoretical point of view, the inhomogeneous MUSIG model appears to be quite robust and good accuracy can be obtained by using a sufficient amount of velocity groups in addition to the size groups. With enough size classes the exact shape of the PSD can be approximated very reasonably. Unfortunately, in the CFD setting the computational costs of solving multiple momentum equations are quite high and in practice it may be feasible to use only a few, ie. 2 or 3, different velocity groups. This is naturally quite a substantial limitation and it takes away large part of the flexibility of the method. The limited amount of velocity groups would be especially problematic in cases, where the PSD is largely spatially dependent, as it would be hard to choose the velocity groups so that they represent the PSD satisfactory in all parts of the domain.

Using many size groups per velocity group, ie. more scalar equations, would be relatively cheap, but the accuracy gains in heavily size dependent flows is questionable. In fact, if we assume that there are no size transfer mechanisms, meaning the source terms are zero, the continuity equations for each particle size within each velocity group will be identical. So using more than one size group within a velocity group would not contribute anything and the accuracy of the solution can be improved only by using more velocity groups.

Despite the problems mentioned, the inhomogeneous MUSIG can still be considered as a serious alternative for modelling PBEs also in fluidized beds, although currently the computational requirements can be a bit prohibitive. Information about the implementation details of MUSIG can be found in the paper by Shi et al. [40] and in the presentation by Frank et al. [41] and some recent developments and extensions are discussed by Krepper et al. [42] and Lucas et al. [43]

Since the aim of this thesis is to find a fast method to take account the PSD, something faster than the class methods would be preferable. The method of moments, which is presented in the next section, tries to reduce the required workload by considering only the moments of the size distribution. The idea is, that it should be possible to obtain information about the distribution easier from its moments than from the rather arbitrary size classes.

4.3 The quadrature method of moments

The method of moments (MOM) was first introduced by Hulburt and Katz in 1964 and it forms the basis for many promising solution strategies. [44] The basic idea of MOM is to transform the original PBE into a problem of finding the moments of the size distribution. The integer moments of a distribution are defined as

$$m^k(\mathbf{x},t) = \int f(\mathbf{x},\xi,t)\xi^k \,\mathrm{d}\xi, \tag{63}$$

in which the integration is over the whole domain where ξ is defined. In principle, one should know all the moments of a distribution in order to define it precisely, but in many applications already the first few moments can give sufficient information about the distribution. For example, the first moment gives the total number of particles:

$$m^0(\mathbf{x},t) = \int f(\mathbf{x},\xi,t) \,\mathrm{d}\xi = N \tag{64}$$

and assuming that the ξ represents particle diameter the next three moments are relative to the total length, surface area and volume of the particles. For fluid dynamic considerations the important Sauter mean diameter can be obtained from knowing the third and fourth moments, as the continuous equivalent for the equation (8) can be expressed as:

$$SMD = \frac{\int f(\mathbf{x},\xi,t)\xi^3 \,\mathrm{d}\xi}{\int f(\mathbf{x},\xi,t)\xi^2 \,\mathrm{d}\xi} = \frac{m^3}{m^2}. \tag{65}$$

The PBE can be transformed to moment form by multiplying it with a some power of ξ and integrating over the whole interval. This way a scalar continuity equation can be obtained for each moment. However, the equations contain unclosed integral terms and as such they are not useful for numerical solution.

A practical approach to solve the closure problem is the quadrature method of moments (QMOM) introduced by McGraw. [45] In the QMOM, the integral terms containing the size distribution function are approximated by numerical quadratures in the same fashion as in numerical integration. With this approach, for example the moments can be expressed as

$$m^k(\mathbf{x},t) \approx \sum_{i=0}^{N} w_i \xi_i^k, \tag{66}$$

where w_i are the weights and ξ_i the abscissas of the quadrature.

The essence of the quadrature approximation is the fact that the weights and the abscissas can be calculated if the values of the moments are known. There are a number of algorithms for this, but a popular choice is the so called product-difference algorithm by Gordon. [46] As the description of the PD algorithm is a bit involved, it is presented separately in section 4.5. Generally it is possible to determine $N/2$ weights and abscissas from N known moments.

It the QMOM approach at the beginning of a simulation, a selected amount of moments are calculated from the initial PSD. After this, the evolution of each moment is tracked with a scalar continuity equation and whenever the weights and abscissas are needed, they are determined from the moments with the PD algorithm. Overall the solution process is quite similar to inhomogeneous class methods and indeed, it is possible to associate the weights and abscissas to the size groups and consider them to form a delta-function approximation for the PSD. Nevertheless, there are still important differences between the two approaches. In MOM, the weights and abscissas are obtained automatically by forcing them to give the right moments, while in the CM the size groups have to be preselected. In MOM the positions of the abscissas can also change constantly along the solution process, while in the CM the reference sizes are fixed. On the other hand, in CM the shape of the PSD can be obtained easily, but with the moment approximation this information is at least partially lost.

The QMOM approach has given promising results in various applications and it has been also successfully implemented in the CFD context by Marchisio et al. [47–49] In a comparison of a two-dimensional bubbly column simulation made by Sanyal et al., it was found out that in general both the QMOM and (homogeneous) class methods give similar results. However, in the QMOM it was sufficient to track only the first 6 moments, while in the CM 12 size groups were required for the same accuracy. [50] Similar results have also been obtained in other studies, see eg. Silva et al. [51] and Selma et al. [52], and generally it appears, that the moment approach could substantially reduce the amount of scalar equations needed for PBE modelling.

Unfortunately, the QMOM shares the same drawback as the homogeneous class methods, ie. the limitation to single particle velocity field. In QMOM all the moments propagate with the same velocity and there is no mechanism to account for size dependent velocities. Technically each moment could have a separate velocity field, but there is no straightforward way to calculate those. Fortunately, since the introduction of QMOM, improved models have been developed which take care of this issue. One of the most promising methods is the DQMOM, which is presented in the next section.

4.4 The direct quadrature method of moments

The direct quadrature method of moments (DQMOM) is a modification of the QMOM originally proposed by Marchisio and Fox. [53] The idea of the DQMOM is to write transport equations directly for the weights and abscissas appearing in the quadrature approximation instead of tracking the moments themselves. The amount of scalar equations is the same as in the QMOM when considering similar amount of moments and Marchisio and Fox have shown that for monovariate cases the DQMOM and QMOM are equivalent. The biggest benefit of the DQMOM compared to the QMOM is the possibility to use separate velocity fields for each of the transported weights and abscissas thus allowing different transportation behaviour for different particle sizes. DQMOM is also easier to apply to multivariate PBEs, which are hard to use with the regular QMOM.

In the derivation of the DQMOM the PSD is first written in terms of delta-functions

$$f(\mathbf{x},L,t)=\sum_{q=1}^{N} w_q(\mathbf{x},t)\delta[L-L_q], \tag{67}$$

where w_q and L_q are the weights and abscissas of the quadrature approximation. The delta-function approximation is then substituted into the PBE and after some manipulations, transport equations for each weight and abscissa are obtained. The equations can be written as

$$\frac{\partial w_q}{\partial t}+\nabla\cdot(\mathbf{v}_q w_q)=a_q \tag{68}$$

$$\frac{\partial \mathscr{L}_q}{\partial t}+\nabla\cdot(\mathbf{v}_q \mathscr{L}_q)=b_q, \tag{69}$$

where $\mathscr{L}_q = L_q w_q$ is the weighted abscissa. In the equations the velocities $\mathbf{v}_q$ are now specific to each weight and abscissa and no homogeneous velocity assumption is needed. The terms a and b are relative to the source terms of the PBE and they are determined from a set of linear equations. By combining the terms into a single vector α

$$\alpha=[a_1\cdots a_N b_1\cdots b_N],$$

the linear system can be expressed as

$$\mathbf{A}\alpha=d. \tag{70}$$

The coefficient matrix A has the form

$$\mathbf{A}=[A_1A_2],$$

where

$$\mathbf{A}_1=\begin{pmatrix} 1 & \cdots & 1 \\ 0 & \cdots & 0 \\ -L_1^2 & \cdots & -L_N^2 \\ \vdots & \ddots & \vdots \\ -2(1-N)L_1^{2N-1} & \cdots & -2(1-N)L_N^{2N-1} \end{pmatrix} \tag{71}$$

$$\mathbf{A}_2 = \begin{pmatrix} 0 & \cdots & 0 \\ 1 & \cdots & 1 \\ 2L_1^2 & \cdots & 2L_N^2 \\ \vdots & \ddots & \vdots \\ (2N-1)L_1^{2N-2} & \cdots & (2N-1)L_N^{2N-2} \end{pmatrix} \tag{72}$$

The term d is defined as

$$d = [\bar{S}_0 \cdots \bar{S}_{2N-1}], \tag{73}$$

where

$$\bar{S}_k(\mathbf{x},t) = \int_0^\infty L^k S(\mathbf{x},L,t)\,\mathrm{d}L. \tag{74}$$

The linear system has to be solved in each iteration and in each control volume.

The first DQMOM implementation for gas-solid flows was presented by Fan et al. [54] In the implementation, the solid volume fraction is expressed as a function of the weights and abscissas:

$$\alpha_q = k_v L_q^3 w_q \tag{75}$$

$$\alpha_q L_q = k_v L_q^4 w_q \tag{76}$$

where the k_v is the volumetric shape factor, which is $\pi/6$ for spherical particles. When replacing the weights with the solid volume fraction, the equations (68)-(69) can be written as

$$\frac{\partial \alpha_q \rho_q}{\partial t} + \nabla \cdot (\mathbf{v}_q \alpha_q \rho_q) = 3k_v \rho_q L_q^2 b_q - 2k_v \rho_q L_q^3 a_q \tag{77}$$

$$\frac{\partial \alpha_q L_q \rho_q}{\partial t} + \nabla \cdot (\mathbf{v}_q \alpha_q L_q \rho_q) = 4k_v \rho_q L_q^3 b_q - 3k_v \rho_q L_q^4 a_q. \tag{78}$$

The equation (77) has now the exact form of a multiphase continuity equation (17) and it can be coupled with a corresponding momentum equation. This way the particles can be presented as a set of distinct solid phases, which each has a different diameter L_q, volume fraction α_q and velocity $\mathbf{v}_q$ and all the equations are solved together with the equations for the fluid phase. This presentation is thus similar to that of inhomogeneous MUSIG, except that instead of scalar equations for size groups, there are now scalar equations (78) for the weighted abscissas.

If the particles are inert, as it is assumed in this thesis, the equations (77)-(78) reduce to

$$\frac{\partial \alpha_q \rho_q}{\partial t} + \nabla \cdot (\mathbf{v}_q \alpha_q \rho_q) = 0 \tag{79}$$

$$\frac{\partial \alpha_q L_q \rho_q}{\partial t} + \nabla \cdot (\mathbf{v}_q \alpha_q L_q \rho_q) = 0, \tag{80}$$

and there is no need to solve the linear system (70). Also, if the initial condition represents a completely homogeneous mixture and there are no generating boundary conditions, which can also be assumed in the simulation cases presented in this thesis, the abscissas L_q will be constant for all times. This means that the equations (80) are not needed and only one continuity and momentum equation has to be solved for each quadrature point making the DQMOM very simple to implement. Of course in practice, it depends on the simulated case, whether or not these kinds of approximations are suitable.

The DQMOM has been validated in several different applications, including also fluidized beds, and it has given promising results. [51, 52, 54, 55] In homogeneous cases it behaves like the QMOM and good results can be usually obtained with just a few quadrature points, ie. with two to four points. This is very desirable from a computational point of view and it is much less than what is reported for class methods. For these reasons the DQMOM has gained interest and it would seem to be a good candidate for modelling PSDs in fluidized beds. There is the reference implementation by Fan et al. [54], which shows the feasibility of the method and actually a basic version of DQMOM has been implemented as an extension module in the newest version (v14) of FLUENT. However, since the DQMOM and other moment methods are still relatively new, they have not been thoroughly validated especially with fluidized beds and there are some known drawbacks, which should be taken into account.

One drawback is the possibility of numerical problems because of ill-conditioning. Grosch et al. have developed a generalized framework for quadrature based moment methods, from which they are able to derive the standard QMOM, DQMOM and also other quadrature formulations. [56] In their study they found out, that with the usual moment methods the mapping from the moments to the weights and abscissas tends to be quite badly conditioned and the situation gets quickly worse with increasing number of quadratures. So although the moment methods may give good results with just a few quadratures, obtaining additional accuracy by increasing the amount of quadratures may not be practical for numerical reasons.

With DQMOM the ill-conditioning can be observed if two or more abscissas L_q became too close to each other. In these cases the linear system (70) becomes badly conditioned and the matrix A gets singular if two abscissas are identical. If this happens, the solution process can't proceed without some corrections. Fan et al. discuss this problem and they have found out that bad singularities are indeed encountered in realistic simulation cases, although the frequency of them is quite low. [54] In order to recover from the singularities, Fan et al. suggest calculating the condition number of the linear system and in case of singularity, using either zero solution or averaging the solutions from neighbouring computation cells. In both cases the convection in physical space should clear the singularity. Also, if there are no source terms, the linear system doesn't have to be solved at all and the singularities can be avoided.

Another pitfall for some moment methods is the possibility to obtain locally invalid moment sets, ie. sets of moments, which do not correspond to any real PSD. The values of the moments in a moment set have to satisfy certain conditions in order to represent a physically realistic PSD. If these conditions are not met, the resulting weights and abscissas will not be reasonable and contain for example negative values. Invalid moment sets can be encountered with those moment methods which track the moments but allow separate velocity fields for them.

The issue of invalid moment sets and the reasons leading to them are comprehensively presented by Wright. [57] To summarize the findings, the main culprit is in the discretization of the advection term in the moment transport equations, especially when second or higher order schemes are used. The discretization schemes are designed to transport independent variables without having to preserve relations among them. When the moments are advected, even if the errors for each separate moment are small, the ratios between the moments in a computational cell can be substantially altered and cause the set to become invalid. The first order upwind tends to avoid this problem, but it may be too diffusive for practical computations and according to a very recent study by Mazzei et al., even then the numerical errors and other sources can lead to invalid moment sets. [58]

The numerical tests performed by Wright and Mazzei et al. show, that although the moment sets may be invalid only quite infrequently, problems still occur in practice. Regardless of the cause, if an invalid moment set is encountered, some kind of a recovery procedure is needed to continue the simulations. Some suggestions for recovery are given for example by Wright and Mazzei et al.

Since in the DQMOM the weights and abscissas are directly tracked and the moments are calculated from them, instead of vice versa, the method is fortunately unaffected by the invalid moments issue as such. However, it still possible to obtain unrealistic weights and abscissas in some cases. In a recent paper by Mazzei et al., the effect of numerical diffusion to the DQMOM was studied. [59] In their paper, Mazzei et al. tried to simulate particle mixing in a simple, small scale 2D fluidized bed. Initially the bed had two layers of particles with clearly different diameters. The smaller particles were on the bottom and larger on the top. According to real experiments, the particles should quickly mix and create a homogeneous mixture when the fluidizing air is turned on.

The case was simulated using two quadratures and the particles were assumed to be inert. The DQMOM was able to predict the mixing, and the relative weights corresponding to the two particle sizes were correct in the mixture. However, the locations of the abscissas had moved close to each other in between the small and large particle diameters as if there had been a change in the particle sizes. According to Mazzei et al., the reason for this was the numerical diffusion, which affects the solution of the DQMOM equations. If a diffusion term is added to the PBE, the resulting DQMOM equations will have source terms even with inert particles. These source terms would prevent the non-realistic drift of the abscissas and they should be included even in the case of inert particles. If only the standard DQMOM equations (79)-(80) are used, the numerical diffusion is uncompensated and the particle system will have size changing behaviour.

Mazzei et al. were able to correct the errors in the abscissas and obtain good results by estimating the amount of numerical diffusion and adding the corresponding source terms to the DQMOM equations. However, in general this kind of correction can be tricky to apply, because the amount of numerical diffusion depends on many parameters and can be different in each computational cell. Also, according to Mazzei et al., the simulations were quite sensitive to the estimated value of diffusion, which further complicates the corrections.

On the other hand, typically the amount of numerical diffusion is not very substantial and if there are actual mass transfer mechanisms, the source terms related to those may well be large enough to make the effect of numerical diffusion negligible. Mazzei et al. state that this is probably the reason, why the diffusion problem has not been encountered in studies with mass transfer effects. Also, if the initial distribution is well mixed, so that the abscissas are uniform in the whole domain, the source terms resulting from diffusion will be identically zero. In these cases the numerical diffusion poses no extra difficulties.

Despite the presented drawbacks, in general the DQMOM can still be considered as a promising and useful approach as long as the limitations and possible caveats are taken into consideration. However, even though the DQMOM should be reasonably fast compared to for example class methods, every additional momentum equation will make the simulations much more computationally demanding. This is not a very desirable property, as the multiphase simulations are already very time consuming. Since one of the main motivations for this thesis was to find out, if the PSD could be modelled with only little additional effort, it would be preferable to avoid too much extra computations.

In principle there is no need to solve multiple momentum equations if the velocities for the different particle sizes could be obtained some other way. This idea leads to a mixture approach, in which the particle momentum equation is considered to represent the mixture of the particles and the individual velocities for the different size classes are solved from algebraic force balance equations. This idea is presented in the next chapter.

4.5 The product-difference algorithm

The product-difference (PD) algorithm is a method to compute the abscissas L_i and weights w_i which are required in the quadrature approximation of a PSD. The algorithm needs $2N$ first moments ($m_0 \ldots m_{2N-1}$) of the distribution as an input to produce N weights and abscissas. In the algorithm a tridiagonal matrix is constructed recursively from the moments and the weights and abscissas are obtained from the eigenvalues and the eigenvectors of this matrix. The PD method was first presented by Gordon [46] and the notation used here follows the one used by McGraw [45].

The procedure starts by defining a $(2N+1) \times (2N+1)$ sized array P, where n is the amount of quadrature points. The values of the first column are

$$P(i,1) = \delta_{i,1} \qquad i \in 1 \ldots 2N+1, \tag{81}$$

where $\delta_{i,1} = 1$ for $i = 1$ and $\delta_{i,1} = 0$ for $i \neq 1$. The second column contains the moments with alternating signs:

$$P(i,2) = (-1)^{i-1} m_{i-1} \qquad i \in 1 \ldots 2N. \tag{82}$$

To make the calculations simpler, it can be assumed here that the distribution is normalized, ie. $m_0 = 1$. The true weights can be obtained by multiplying the results with the correct m_0. The other columns can be filled with the recursion formula

$$P(i,j) = P(1,j-1)P(i+1,j-2) - P(1,j-2)P(i+1,j-1), \tag{83}$$

where $j \in 3 \ldots 2N+1$ and $i \in 1 \ldots 2N+2-j$. The equation (83) is what gives the PD algorithm its name. As an example, for the case of $n = 2$, one should get an array like

$$P(i,j) = \begin{pmatrix} 1 & 1 & m_1 & m_2 - m_1^2 & m_1 m_3 - m_2^2 \\ 0 & -m_1 & -m_2 & -m_3 + m_1 m_2 & 0 \\ 0 & m_2 & m_3 & 0 & 0 \\ 0 & -m_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{84}$$

After obtaining the array P, a n-dimensional vector α can be defined as:

$$\alpha(j) = \begin{cases} 0 & \\ \frac{P(1,j+1)}{P(1,j)P(1,j-2)} & j \in 2 \ldots 2N. \end{cases} \tag{85}$$

With this vector we can calculate the diagonal a_j and co-diagonal b_j components of a symmetric tridiagonal matrix:

$$\begin{cases} a_j &= \alpha(2j) + \alpha(2j-1) \\ b_j &= \sqrt{\alpha(2j)\alpha(2j-1)}. \end{cases} \tag{86}$$

with $j \in 1 \ldots N$. The resulting matrix

$$\mathbf{J}_n = \begin{pmatrix} a_1 & b_1 & & \\ b_1 & a_2 & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{pmatrix} \tag{87}$$

is also known as Jacobi matrix.

Finally, to get the weights and abscissas, one has to calculate the eigenvalues λ and eigenvectors v of the matrix $\mathbf{J}$. Since $\mathbf{J}$ is symmetric and tridiagonal, this should be easy and the eigenvalues are real. The abscissas are directly the eigenvalues,

$L_i = \lambda_i$, and the weights are obtained from the first components of the respective eigenvectors:

$$w_j = m_0 v_{j1}^2. \tag{88}$$

By using here the actual m_0, one gets the non-normalized weights. If all the steps were performed correctly, the N generated weights and abscissas should give correctly the first $2N$ moments when applied in the quadrature approximation.

5 Implementation

5.1 The general idea of the approach

In the previous chapter some of the various methods for including a PSD in multiphase flow simulation were presented and their applicability for fluidized bed modelling was investigated. Of the several possible alternatives, the DQMOM approach seemed to be most suitable for fluidized bed applications. The DQMOM allows separate velocity fields for differently sized particles and relatively good accuracy could be expected by including only two or three different particle sizes in the simulations. There are also existing implementations available, which would make it easy to adopt the method. However, in the usual implementations of the DQMOM, for example in the implementation by FLUENT, the differently sized particles are represented as distinct solid phases with separate momentum equations. This is computationally very demanding and not ideal for any larger scale modelling.

Besides the full Eulerian-Eulerian flow equations (17)-(19), fluid-particle flows can also be modelled with the so called mixture approach. The mixture approach is a simplified version of the Eulerian-Eulerian model, in which only a single momentum equation for the whole fluid-particle mixture is solved. In the mixture approach it is assumed, that the fluid and solid phases are strongly coupled and that at least on a local scale, there is force equilibrium between the phases. This assumption allows to formulate an algebraic relation between the velocities of the solid and fluid phases and no separate momentum equation is needed for the solid phase. In addition to the mixture equation, only continuity equations are needed for each phase. A description and derivation for the mixture model is presented for example in the report by Manninen et al. [60]

The mixture model is much quicker to compute than the full multiphase model, but unfortunately, the assumption of local equilibrium and strong coupling between the solid particles and the fluid is typically not very good in fluidized bed applications and for this reason, the full multiphase equations are usually employed. However, instead of assuming equilibrium between the fluid and the solid phases, it is possible to restrict the mixture approach to the solid particles only. If the differently sized particles are presented as separate solid phases, it is possible to consider them to form a mixture and similarly to the usual mixture approach, a mixture momentum equation can be formulated. Also in the similar way, the velocities for the different size classes can be solved from algebraic equations, instead of resorting to separate momentum equations. By formulating the mixture idea this way, it should be possible to get reasonable results, since less restrictive assumptions are needed than those that are required in the classical mixture model.

The derivation of the solid mixture approach with all the required assumptions is given in this chapter. The approach models the PSD with the DQMOM equations as presented in the previous chapter, but instead of separate momentum equations, the velocities for each individual particle size class are determined from a non-linear system of algebraic equations. At least in theory, this approach should be faster than solving momentum equations, which are partial differential equations.

5.2 The mixture formulation

The development of the mixture formulation presented here closely follows Manninen et al., in which the mixture model is derived in the classical sense, ie. with a mixture equation for the whole suspension. [60]

As a starting point for the mixture approach, the full multiphase flow equations are written for each particle size class, as if they were separate solid phases. The continuity and momentum equations (17) and (19) have to be modified to accommodate multiple solid phases. With n different solid phases, the equations can be written as

$$\frac{\partial}{\partial t}(\alpha_q\rho_q)+\nabla\cdot(\alpha_q\rho_q\mathbf{v}_q)=0 \tag{89}$$

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_q\rho_q\mathbf{v}_q)+\nabla\cdot(\alpha_q\rho_q\mathbf{v}_q\mathbf{v}_q)=&-\alpha_q\nabla p-\nabla p_q+\nabla\cdot\tau_q+\alpha_q\rho_q\mathbf{g}\\&+K_{\mathrm{g}q}(\mathbf{v}_{\mathrm{g}}-\mathbf{v}_q)+\sum_{p\neq q}^{n}K'_{pq}(\mathbf{v}_p-\mathbf{v}_q),\end{aligned} \tag{90}$$

where K'_{pq} represents the drag forces between the solid phases p and q. For the drag coefficient there is the well known expression

$$K'_{pq}=\frac{3(1+e_{pq})\left(\frac{\pi}{2}+C_{\mathrm{fr}}\frac{\pi^2}{8}\right)\alpha_p\rho_p\alpha_q\rho_q(d_p+d_q)^2g_{0,\mathrm{pq}}}{2\pi(\rho_pd_p^3+\rho_qd_q^3)}|\mathbf{v}_p-\mathbf{v}_q|, \tag{91}$$

where e_{pq} is the coefficient of restitution and C_{fr} is the coefficient of friction between the two solid phases. [61]

The gas phase flow equations will have to be modified as well to include summation over the drag terms over the solid phases:

$$\frac{\partial}{\partial t}(\alpha_{\mathrm{g}}\rho_q)+\nabla\cdot(\alpha_q\rho_{\mathrm{g}}\mathbf{v}_{\mathrm{g}})=0 \tag{92}$$

$$\frac{\partial}{\partial t}(\alpha_{\mathrm{g}}\rho_{\mathrm{g}}\mathbf{v}_{\mathrm{g}})+\nabla\cdot(\alpha_{\mathrm{g}}\rho_{\mathrm{g}}\mathbf{v}_{\mathrm{g}}\mathbf{v}_{\mathrm{g}})=-\alpha_{\mathrm{g}}\nabla p+\nabla\cdot\tau_{\mathrm{g}}+\alpha_{\mathrm{g}}\rho_{\mathrm{g}}\mathbf{g}+\sum_{q=1}^{n}K_{\mathrm{g}q}(\mathbf{v}_q-\mathbf{v}_{\mathrm{g}}) \tag{93}$$

Now, in the mixture approach the idea is to avoid solving the momentum equations (90) for the individual phases, but instead the contribution from all the solid phases are combined in the single mixture continuity and momentum equations. The equations for the mixture can be obtained by simply summing over all the particle phases. The continuity equation for the mixture can be therefore written as

$$\frac{\partial}{\partial t}\sum_{q=1}^{n}(\alpha_q\rho_q)+\nabla\cdot\sum_{q=1}^{n}(\alpha_q\rho_q\mathbf{v}_q)=o. \tag{94}$$

By assuming uniform density for the particles, the continuity equation can be simplified a bit:

$$\frac{\partial}{\partial t}(\alpha_{\mathrm{s}}\rho_{\mathrm{s}}) + \nabla\cdot(\alpha_{\mathrm{s}}\rho_{\mathrm{s}}\mathbf{v}_{\mathrm{s}}) = 0, \tag{95}$$

where the mixture volume fraction and velocity are defined as

$$\alpha_s = \sum_{q=1}^{n} \alpha_q \tag{96}$$

$$\mathbf{v}_{\mathrm{s}} = \frac{\sum_{q=1}^{n} \alpha_q \mathbf{v}_q}{\alpha_{\mathrm{s}}}. \tag{97}$$

Since the equation (95) has the same form as the continuity equation for a single solids phase (17), the subscript s is used also here to represent the particle mixture.

The momentum equation for the particle mixture follows also from summing over the phases:

$$\begin{aligned}\frac{\partial}{\partial t}\sum_{q=1}^{n}(\alpha_q\rho_q\mathbf{v}_q) + \nabla\cdot\sum_{q=1}^{n}(\alpha_q\rho_q\mathbf{v}_q\mathbf{v}_q) = &-\sum_{q=1}^{n}\alpha_q\nabla p - \sum_{q=1}^{n}\nabla p_q + \nabla\cdot\sum_{q=1}^{n}\tau_q \\ &+ \sum_{q=1}^{n}\alpha_q\rho_q\mathbf{g} + \sum_{q=1}^{n}K_{\mathrm{g}q}(\mathbf{v}_{\mathrm{g}} - \mathbf{v}_q) \\ &+ \sum_{q=1}^{n}\sum_{p\neq q}^{n}K'_{pq}(\mathbf{v}_p - \mathbf{v}_q).\end{aligned} \tag{98}$$

The equation looks quite complicated, but fortunately it can be simplified a bit. Using the definition of the mixture velocity, the second term in the equation (98) can be written as

$$\nabla\cdot\sum_{q=1}^{n}\alpha_q\rho_q\mathbf{v}_q\mathbf{v}_q = \nabla\cdot(\alpha_{\mathrm{s}}\rho_{\mathrm{s}}\mathbf{v}_{\mathrm{s}}\mathbf{v}_{\mathrm{s}}) + \nabla\cdot\sum_{q=1}^{n}\alpha_q\rho_q\mathbf{v}_{q\mathrm{s}}\mathbf{v}_{q\mathrm{s}}, \tag{99}$$

where $\mathbf{v}_{q\mathrm{s}}$ is the slip velocity

$$\mathbf{v}_{q\mathrm{s}} = \mathbf{v}_q - \mathbf{v}_{\mathrm{s}}. \tag{100}$$

The mixture stress and the diffusive stress due to velocity slip can be defined as

$$\tau_{\mathrm{s}} = \sum_{q=1}^{n}\tau_q \tag{101}$$

$$\tau_{\mathrm{Ds}} = -\sum_{q=1}^{n}\alpha_q\rho_q\mathbf{v}_{\mathrm{s}q}\mathbf{v}_{\mathrm{s}q} \tag{102}$$

and the mixture solids pressure as

$$p_{\mathrm{s}} = \sum_{q=1}^{n} p_q. \tag{103}$$

With these definitions and by noticing that the double sum over the solid-solid drag terms will vanish, the mixture momentum equation takes the form

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_s\rho_s\mathbf{v}_s)+\nabla\cdot(\alpha_s\rho_s\mathbf{v}_s\mathbf{v}_s)=&-\alpha_s\nabla p-\nabla p_s+\nabla\cdot(\tau_s+\tau_{Ds})+\alpha_s\rho_s\mathbf{g}\\&+\sum_{q=1}^{n}K_{gq}(\mathbf{v}_g-\mathbf{v}_q).\end{aligned}\tag{104}$$

The velocity slip equation can be derived by combining the momentum equations of a single solid phase (90) and the solid mixture (104). To simplify the computations, the left hand sides of the both equations are first written in the so called non-conservative form. In general, by using the continuity equation it can be shown that

$$\frac{\partial}{\partial t}(\alpha\rho\mathbf{v})+\nabla\cdot(\alpha\rho\mathbf{v}\mathbf{v})=\alpha\rho\frac{\partial}{\partial t}(\mathbf{v})+\alpha\rho(\mathbf{v}\cdot\nabla)\mathbf{v}.\tag{105}$$

In the non-conservative form the equations (90) and (104) are thus

$$\begin{aligned}\alpha_q\rho_q\frac{\partial}{\partial t}(\mathbf{v}_q)+\alpha_q\rho_q(\mathbf{v}_q\cdot\nabla)\mathbf{v}_q=&-\alpha_q\nabla p-\nabla p_q+\nabla\cdot\tau_q+\alpha_q\rho_q\mathbf{g}\\&+K_{gq}(\mathbf{v}_g-\mathbf{v}_q)+\sum_{p\neq q}^{n}K'_{pq}(\mathbf{v}_p-\mathbf{v}_q)\end{aligned}\tag{106}$$

$$\begin{aligned}\alpha_s\rho_s\frac{\partial}{\partial t}(\mathbf{v}_s)+\alpha_s\rho_s(\mathbf{v}_s\cdot\nabla)\mathbf{v}_s=&-\alpha_s\nabla p-\nabla p_s+\nabla\cdot(\tau_s+\tau_{Ds})+\alpha_s\rho_s\mathbf{g}\\&+\sum_{q=1}^{n}K_{gq}(\mathbf{v}_g-\mathbf{v}_q).\end{aligned}\tag{107}$$

By solving the pressure gradient from the latter equation and inserting it to the former, after some simplifications and assuming $\rho_q=\rho_s$ the expression

$$\begin{aligned}K_{gq}(\mathbf{v}_g-\mathbf{v}_q)+\sum_{p\neq q}^{n}K'_{pq}(\mathbf{v}_p-\mathbf{v}_q)=&\alpha_q\Big(\rho_q\frac{\partial}{\partial t}\mathbf{v}_{qs}+\rho_q((\mathbf{v}_q\cdot\nabla)\mathbf{v}_q-(\mathbf{v}_s\cdot\nabla)\mathbf{v}_s)\\&+\nabla\left(\frac{p_q}{\alpha_q}-\frac{p_s}{\alpha_s}\right)+\frac{\alpha_q}{\alpha_s}\sum_{q=1}^{n}K_{gq}(\mathbf{v}_g-\mathbf{v}_q)\\&+\frac{\nabla\cdot(\tau_s+\tau_{Ds})}{\alpha_s}-\frac{\nabla\cdot\tau_q}{\alpha_q}\Big)\end{aligned}\tag{108}$$

is obtained.

Up to this point only relatively mild assumptions were required, but to get further, more restrictive approximations have to be made. First by assuming local force equilibrium, the time derivative of the slip velocity will be zero:

$$\frac{\partial}{\partial t}\mathbf{v}_{qs}\approx 0.\tag{109}$$

This is the primary approximation of the mixture approach and it's validity is discussed separately in section 6.6.

Next the convective terms are approximated to be of similar order, ie.

$$(\mathbf{v}_q \cdot \nabla)\mathbf{v}_q \approx (\mathbf{v}_\mathrm{s} \cdot \nabla)\mathbf{v}_\mathrm{s}. \tag{110}$$

In order to get rid of the solids pressure, it is approximated that

$$p_q \approx \frac{\alpha_q}{\alpha_\mathrm{s}} p_\mathrm{s}. \tag{111}$$

The validity of this assumption can be questioned, since for example none of the expressions (21)-(23) for the solid pressure satisfies it. On the other hand, the assumption is consistent in a situation, where the solid particles are represented with n identical phases, whereas the other expressions are not. The inconsistency occurs, because the typical formulations for the solid pressure assume that there is only a single solid phase and they would require modifications to be valid with multiple phases. However, for simplicity, the normal formulations are assumed to hold and the approximation (111) is made.

Finally, it is approximated that the viscous and diffusive terms are small compared to the drag terms and that the difference between the last two terms in the equation (108) is small. For this reason the last two terms are omitted. After all the approximations, what is left is simple force balance statement

$$K_{\mathrm{g}q}(\mathbf{v}_\mathrm{g} - \mathbf{v}_q) + \sum_{p \neq q}^{n} K'_{pq}(\mathbf{v}_p - \mathbf{v}_q) = \frac{\alpha_q}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}), \tag{112}$$

from which the velocities $\mathbf{v}_q$ can be solved for each solid phase after the gas and mixture velocities are calculated. The left-hand side of the relation consists of the drag force between the gas and a solid phase and the sum of all solid-solid drag terms which affect the particular phase. On the right hand side there is the relative portion of the total gas-solids drag for the particular phase. The equation states, that these forces have to be in balance. Also, by summing over all the solid phases it follows that

$$\sum_{q=1}^{n} K_{\mathrm{g}q}(\mathbf{v}_\mathrm{g} - \mathbf{v}_q) + \sum_{q=1}^{n} \sum_{p \neq q}^{n} K'_{pq}(\mathbf{v}_p - \mathbf{v}_q) = \sum_{q=1}^{n} \frac{\alpha_q}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \tag{113}$$

$$\Rightarrow \sum_{q=1}^{n} K_{\mathrm{g}q}(\mathbf{v}_\mathrm{g} - \mathbf{v}_q) = K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}). \tag{114}$$

When written out, the equation (112) forms a system of non-linear equations. If two or three size classes are used, the equations can be written as

$$\begin{cases} K_\mathrm{g1}(\mathbf{v}_\mathrm{g} - \mathbf{v}_1) + K'_{12}(\mathbf{v}_2 - \mathbf{v}_1) = \frac{\alpha_1}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \\ K_\mathrm{g2}(\mathbf{v}_\mathrm{g} - \mathbf{v}_2) + K'_{12}(\mathbf{v}_1 - \mathbf{v}_2) = \frac{\alpha_2}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \end{cases} \tag{115}$$

for two sizes and as

$$\begin{cases} K_\mathrm{g1}(\mathbf{v}_\mathrm{g} - \mathbf{v}_1) + K'_{12}(\mathbf{v}_2 - \mathbf{v}_1) + K'_{13}(\mathbf{v}_3 - \mathbf{v}_1) = \frac{\alpha_1}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \\ K_\mathrm{g2}(\mathbf{v}_\mathrm{g} - \mathbf{v}_2) + K'_{12}(\mathbf{v}_1 - \mathbf{v}_2) + K'_{23}(\mathbf{v}_3 - \mathbf{v}_2) = \frac{\alpha_2}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \\ K_\mathrm{g3}(\mathbf{v}_\mathrm{g} - \mathbf{v}_3) + K'_{13}(\mathbf{v}_1 - \mathbf{v}_3) + K'_{23}(\mathbf{v}_2 - \mathbf{v}_3) = \frac{\alpha_3}{\alpha_s} K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}). \end{cases} \tag{116}$$

for three. Even with three phases these equations are still relatively simple, but if more size classes are used, the equations get increasingly complex.

The non-linear systems have to be solved in each control volume and at each iteration, which does increase the computational costs. On the other hand, with only two or three equations at a time, these systems are very small and solving them should not take very long. Since the equations are non-linear, they have to be solved iteratively, which could be time consuming at least in the worst case. However, since it is likely that the velocities do not change much from one time step to another, the previous solution can be used as a very good initial condition.

5.3 Volume fraction corrections

In general when multiphase flow equations are solved, the adopted solution strategy should ensure the consistency of the volume fractions. This means that the total sum of the volume fractions should be unity and if the mixture approach is used, the sum of the particle phases should also be equal to the mixture volume fraction. These requirements can be stated as

$$\sum_{q=1}^{n} \alpha_q = \alpha_\mathrm{s} \tag{117}$$

$$\alpha_\mathrm{s} + \alpha_\mathrm{g} = 1 \tag{118}$$

and ideally they should be satisfied at every time step during the simulation. With the mixture approach these requirements imply that the sum of the individual size class volume fluxes should be equal to the mixture flux at each point in the domain. However, since the mixture velocity and the velocities for the different particle size classes are only approximate, it is clear that this does not hold if the fluxes are not corrected by some way or another.

With multiphase flows the typical strategy to achieve consistency is to let one volume fraction to be elastic, ie. omit the continuity equation for the corresponding phase and solve its volume fraction from the consistency equations. For example in the Phase Coupled SIMPLE (PC-SIMPLE) approach used in FLUENT, a continuity equation for the total volume conservation is first formed and used as a basis for the pressure-correction equation. After the corrected velocities are attained, the volume fractions of the phases are solved from the phase continuity equations, but this is only done for the dispersed phases. The gas phase volume fraction is simply what is left, or in other words, the unity minus the sum of the solid volume fractions. [1]

With the mixture approach a reasonable strategy would be to make the mixture volume fraction elastic. This way only the DQMOM continuity equations for the size class VOFs would be solved and the mixture and gas volume fractions could be calculated from the consistency equations. However, implementing this idea as such would require modifications to the pressure-velocity coupling scheme. The current formulation assumes that each volume fraction belongs to a phase with a separate momentum equation and this would naturally make the mixture approach useless.

Unfortunately in FLUENT, it is not possible to modify the pressure-velocity coupling and for this reason some other solution has to be used. Luckily the volume fractions can be changed manually via the user defined macros and therefore at least in principle, some kind of a correction procedure could be implemented. It is also possible to toggle off the solid volume fraction calculation altogether but if this is done, the solver stops reacting to the solid volume fraction value and for example the solid pressure does not change from its initial value even if the volume fractions are changed from a macro. Therefore, the volume fraction equation has to be included in every case.

Even though the pressure-velocity coupling can't be modified and the mixture volume fraction has to be calculated from a continuity equation, it is still possible to try to mimic the elastic mixture volume fraction strategy. The mixture VOF can be replaced with the sum of the size class volume fractions in each iteration and this will make it effectively elastic. However, when this strategy was tried, it was found to be very unstable and the solution tended to diverge very quickly. Setting different under-relaxation factors or applying the correction only at every time step instead of at every iteration did not make a difference.

Since it is not possible to easily correct the mixture VOF, the situation can be considered from an opposite viewpoint. Because the mixture velocity should represent the overall solids flow with reasonable accuracy and the size class velocities are only approximated from it, it would be also logical to use the mixture continuity equation to solve the mixture VOF and require that the size classes satisfy the consistency equation (117). The physically correct way to do this would be to modify the velocities of the size classes in such a way, that the sum of their fluxes would be the same as what the mixture flux is. Unfortunately, it is not easy to devise such a correction scheme, because it is not clear how the velocities should be modified. In contrast to the momentum equations, where the pressure term can be corrected, there are no free terms in the force balance equation (112). This means that either the force balance equations should be modified or some approximative strategy should be applied.

After considering various alternatives how to modify the velocities, at the end for simplicity, an approximative scheme was chosen. The idea of the scheme is to simply force the size class volume fractions to conform to the mixture VOF by scaling them locally in each control volume at the end of each time step.

The correction procedure works as follows: first the mixture and size class fluxes are computed normally and the volume fraction fields are advanced. Since the sum of the size class fluxes is not necessarily the same as the mixture flux, the consistency requirement (117) is no longer satisfied. In other words, if the new volume fractions for the phases q are written as $\widetilde{\alpha_q}$, it can be at least locally that

$$\sum_{q=1}^{n} \widetilde{\alpha_q} \neq \alpha_{\mathrm{s}}. \tag{119}$$

Now, in order to satisfy consistency, it is possible to assume that the size class volume fractions are correct with respect to each other, even though their sum is not exactly equal to the mixture VOF. By assuming this, the new, consistent volume fractions are obtained as

$$\alpha_q = \widetilde{\alpha_q} \frac{\alpha_{\mathrm{s}}}{\sum_{q=1}^{n} \widetilde{\alpha_q}} \tag{120}$$

and now

$$\sum_{q=1}^{n} \alpha_q = \alpha_{\mathrm{s}}. \tag{121}$$

However, there is still one problem left. Even though the volume fractions are locally in balance, the scaling may have affected the total amount of different particle sizes. In order to correct this, further scaling is necessary. This can be done by calculating a correction factor for each particle size as

$$C_q = \frac{V_q}{\widetilde{V_q}}, \tag{122}$$

where V_q is the total volume for the phase q after the correction and $\widetilde{V_q}$ is the original, real volume. By multiplying each of the volume fractions with their own correction factors, the total volumes are conserved. Naturally, after this the local consistency is lost again, but it can be retained by normalizing the volume fractions again with equation (120). By doing this iteratively a couple of times, both the local and global consistency will be obtained.

In order to better illustrate the correction procedure, figure 8 shows a simple made-up scenario with two size classes and three 1D control volumes. The leftmost three boxes in the picture represent the initial situation before the flow time step is advanced and the arrows and numbers on the left are the computed fluxes through the control volume faces. As it can be seen from the figure, the sums of the size class fluxes are close to the mixture fluxes but they are not quite the same. The second picture from the left represents the situation after the fluxes are applied and because the mixture and size class fluxes were not in balance, the sum of the volume fractions are no longer equal to the mixture VOF.

In the third picture the size class volume fractions are scaled in each of the control volumes using the equation (120) and now consistency is attained again locally. However, when the volume fractions are summed over all the control volumes, it can be seen that there is a change in the total amounts, which has to be further corrected.

The last two images represent one correction-normalization iteration. As can be seen from the figure, although the relative errors in fluxes were quite large at the

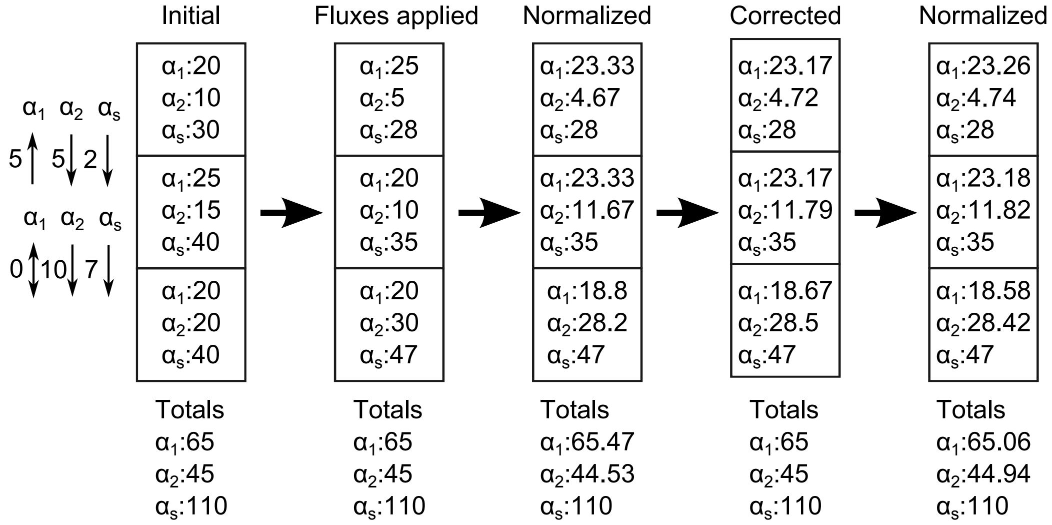

Figure 8: An example describing the volume fraction correction procedure.

beginning, after just one full correction step the situation is quite closely consistent both locally and globally. Although not shown in the figure, if two more correction steps are performed, the relative errors in the global values are of order 10^{-5}.

It is clear that the adopted approximation scheme is far from ideal and its effect on the solution is hard to estimate. While the local normalization (120) is somewhat easy to justify, at least the global correction is definitely something, that is not very preferable. The global scaling can in principle cause unphysical mass transfer where material is moved from one place to other instantly. On the other hand, even if some amount of error is made at each time step, it is difficult to say say what the average error will be since errors can be made in both directions.

When the approximation scheme was tested in the simulations, it was found out that in general, the amount of global correction was not very large. Typically the correction factors C_q were on the order of 10^{-5} or less in the first step and only two or three steps were required to keep the simulation consistent. The simulations could be run even without the global corrections for some time but then there is a risk that at some point the relative amount of the particles will change too much and start to significantly affect the results.

So, even with the known deficiencies of the approach, due to lack of any better method and because of the global correction seemed to remain at small levels, the approximative scheme was employed. Naturally it is recognized that some other, better method should be used if possible.

5.4 Implementation details

The mixture formulation and the DQMOM were implemented in FLUENT by using the user defined functions (UDF). This section provides a brief summary of the implementation assuming that the reader is familiar with the UDF-concept in FLUENT. Detailed information about the UDFs and the specifications for the various macros can be found in the FLUENT manual. [1]

The continuity and momentum equations for the particle mixture were included by defining one Eulerian, granular solid phase. To simplify the implementation, a closure approach was adopted, where instead of using the definition (101) for the mixture stress tensor τ_s, the mixture stress was calculated straight with the mixture variables as

$$\tau_\mathrm{s} = \alpha_\mathrm{s}\mu_\mathrm{s}(\nabla\mathbf{v}_\mathrm{s} + \nabla\mathbf{v}_\mathrm{s}^T) + \alpha_\mathrm{s}(\lambda_\mathrm{s} - \frac{2}{3}\mu_\mathrm{s})(\nabla\cdot\mathbf{v}_\mathrm{s})\mathbf{I}. \tag{123}$$

The mixture momentum equation contains also the diffusive stress term τ_Ds. In principle the diffusive term could be included by defining it as an additional source term to the momentum equation. However, the term contains derivatives of the size class velocities v_q, which are defined as user defined memory locations. In FLUENT the derivatives are only available for user defined scalars and therefore the implementation would require some workarounds. For simplicity, in this thesis the diffusive stress term was assumed to be relatively small and it was omitted to simplify the implementation. However, the importance of the diffusive term is a bit unclear and it would be interesting to see what effect it has on the results.

It was assumed that initially the particle mixture is homogeneous, with no gradients in the quadrature abscissas. This allowed to omit the equations (80) for the abscissas and the numerical diffusion problem discussed in section 4.4 was avoided. To summarize, the equations that are solved in the implementation are

$$\frac{\partial}{\partial t}(\alpha_\mathrm{g}\rho_\mathrm{g}) + \nabla\cdot(\alpha_\mathrm{g}\rho_\mathrm{g}\mathbf{v}_\mathrm{g}) = 0 \tag{124}$$

$$\frac{\partial}{\partial t}(\alpha_\mathrm{g}\rho_\mathrm{g}\mathbf{v}_\mathrm{g}) + \nabla\cdot(\alpha_\mathrm{g}\rho_\mathrm{g}\mathbf{v}_\mathrm{g}\mathbf{v}_\mathrm{g}) = -\alpha_\mathrm{g}\nabla p + \nabla\cdot\tau_\mathrm{g} + \alpha_\mathrm{g}\rho_\mathrm{g}\mathbf{g} + K_\mathrm{gs}(\mathbf{v}_\mathrm{s} - \mathbf{v}_\mathrm{g}) \tag{125}$$

$$\frac{\partial}{\partial t}(\alpha_\mathrm{s}\rho_\mathrm{s}) + \nabla\cdot(\alpha_\mathrm{s}\rho_\mathrm{s}\mathbf{v}_\mathrm{s}) = 0 \tag{126}$$

$$\frac{\partial}{\partial t}(\alpha_\mathrm{s}\rho_\mathrm{s}\mathbf{v}_\mathrm{s}) + \nabla\cdot(\alpha_\mathrm{s}\rho_\mathrm{s}\mathbf{v}_\mathrm{s}\mathbf{v}_\mathrm{s}) = -\alpha_\mathrm{s}\nabla p - \nabla p_\mathrm{s} + \nabla\cdot\tau_\mathrm{s} + \alpha_\mathrm{s}\rho_\mathrm{s}\mathbf{g} + K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \tag{127}$$

$$\frac{\partial\alpha_q\rho_q}{\partial t} + \nabla\cdot(\mathbf{v}_q\alpha_q\rho_q) = 0, \qquad q = 1\ldots n \tag{128}$$

$$K_{\mathrm{g}q}(\mathbf{v}_\mathrm{g} - \mathbf{v}_q) + \sum_{p\neq q}^{n} K'_{pq}(\mathbf{v}_p - \mathbf{v}_q) = \frac{\alpha_q}{\alpha_s}K_\mathrm{gs}(\mathbf{v}_\mathrm{g} - \mathbf{v}_\mathrm{s}) \quad q = 1\ldots n \tag{129}$$

The diameter of the solids was the Sauter mean, which was calculated locally from the weights and abscissas within each control volume. The diameter was specified with a DEFINE_PROPERTY-macro. Each of the weights was defined as a scalar variable and for each of them a scalar transport equation was solved. The default formulations for the convective and time dependent terms were overridden with DEFINE_UDS_FLUX- and DEFINE_UDS_UNSTEADY-macros to be compatible with the DQMOM equations (128).

The velocities needed for the DEFINE_UDS_FLUX-macros were calculated at each iteration by solving the non-linear equations (129). Since the equation systems were very small, ie. two or three equations per control volume, a simple iterative solver using the Newton's method was used. The linearized equations produced by the method were solved with direct Gaussian elimination. In order to keep things simple, no third party libraries were used and both the Newton's method and the Gaussian solver were straightforwardly self-implemented. Since the time step used in the simulations was very small and the velocities changed only little in between the time steps, the solutions typically converged within one or two iterations. For this reason it is unlikely that the non-linear solver would have a big effect on the general performance. However, if maximum performance is desired, it would be possible to use separate, platform optimized solver libraries.

Finally, after each time step the weights were corrected to match the volume fractions by using a DEFINE_EXECUTE_AT_END-macro and following the procedure explained in the section 5.3. The simulations also required using custom boundary conditions, but these are described more in detail in the results chapter. All in all, the implementation of the PSD model and the Newtonian solver and all the boundary value treatments consisted of around 1000 lines of code in total. In addition, about 2000 lines of code were required to calculate the various time averaged values from the transient simulations.

6 Results

6.1 Introduction

In order to test the implemented mixture model, a series of computer simulations were performed. The simulated test cases were based on a laboratory scaled CFB, from which there were controlled experimental results available. The details of the design and construction of the device are presented in Matias Guldén's master's thesis. [62] Information about the device and the previous numerical and experimental studies performed with it can be found in Kallio et al. [63, 64]

In each simulation case the PSD and other relevant parameters were chosen to closely match an actual experimental set-up so that detailed comparisons with experimental results were possible. Also, for each test case a reference simulation was performed in which only the average particle size was used. In order to compare the mixture approach to the standard DQMOM, reference simulations with the FLUENT's own DQMOM implementation would've been also interesting. However, because of the high computational demands of the simulations and because some stability issues were encountered, the reference DQMOM computations were not performed.

It would've been also possible to do the reference simulations with a simple class method -type of an approach, since this is also supported in FLUENT. Unfortunately, in numerical test cases it was noticed, that FLUENT tends to produce unreliable results when multiple solid phases are used. For example, if two identical and initially completely mixed solid phases are used, the phases will quickly segregate even though they should stay mixed. The segregation occurs especially when the amounts of the two phases are not equal. In these cases the phase with a smaller volume fraction will often rise to the top while the other remains at the bottom. Because of this kind of behaviour, it is likely that the results with a PSD would not be accurate.

In the next sections a description of the simulated geometry, the boundary conditions, the common simulation settings and the experimental measurements are given. The validity of the local equilibrium approximation is discussed in section 6.6 and the results from two different test cases are presented in their own sections.

6.2 Geometry description

All the simulation cases used the same computational mesh and geometry, which was taken from the lab-scaled CFB. A picture of the simulated device is shown in figure 9 and in figure 10 the computational domain with the relevant dimensions is presented. The CFB was constructed to be very thin and it has transparent walls to allow optical measurements. The distance between the front- and back walls is only 1.5 cm, which makes the device effectively two dimensional. All the simulations were run as 2D, which allowed to use quite a fine mesh size without excessive computational requirements.

As shown in the figure 10, the solids separator, the return channel and the loop seal were left out of the computational domain and only a small part of the recirculation channel was included. This was done for simplicity reasons, because inclusion of the recirculation parts would've made the simulations much more complicated and accurate modelling of the solids separator would've likely required the use of a 3D mesh. Instead of simulating the recirculating path, the boundary conditions were defined suitably to maintain the amount of solids in the computational domain.

The bulk of the computational domain consisted of the rectangular riser section, which was 3 m tall and 40 cm wide. The riser was divided into uniform, square control volumes with a side length of 5 mm. For the return channel a similarly sized quadrilateral mesh was used. The total amount of computational cells was 49780. The simulation cases were run on a computer cluster and at most 12 cores were used per case. Typically each simulation run took about 15-30 days of computation.

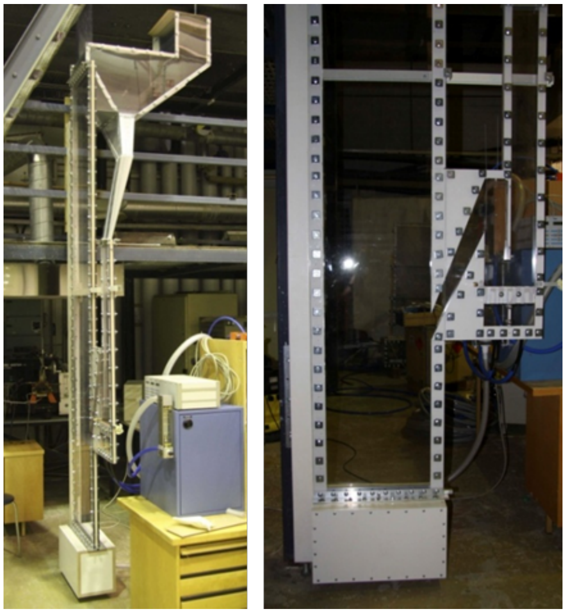

Figure 9: The lab-scale CFB used in the simulations.

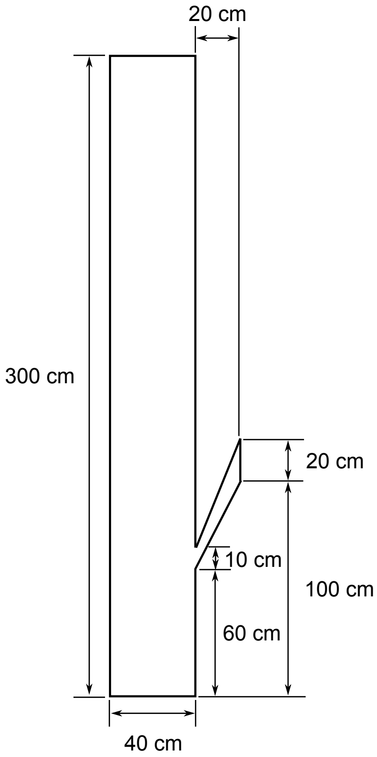

Figure 10: The computational 2D domain used in the simulations. Real thickness of the device is 1.5 cm.

6.3 Boundary conditions

In the computational domain there were four different types of boundaries: the gas inlet at the bottom, the outlet at the top, the solids inlet in the recirculation channel at the side and the walls. Of these four, the outlet and the wall boundaries were relatively straightforward to set up. For the outlet it was natural to use the pressure outlet boundary condition, ie. the pressure value was fixed to ambient value at the boundary. Since with pressure outlet it is possible that the flow direction temporarily reverses, the flow properties should also be given for the back flow. For these cases the returning solids concentration and fluxes for the scalar values were set to zero.

At the walls the no-slip boundary condition was set for the gas phase, which is the typical choice to use. For the solid phase the wall boundary conditions are a bit more complicated, since the solid particles do not stick to the walls as fluids do and the no-slip boundary condition would not be physically correct. One alternative is to use the free slip boundary condition, but this would underestimate the frictional effects of the wall. As a compromise between the no-slip and free slip conditions it is possible to use a partial slip boundary condition.

A quite widely known and often implemented partial slip model was presented by Johnson and Jackson [22] and it uses a specularity coefficient to describe the frictional effects. The value of the coefficient should lie between zero and one, with zero representing the free slip situation. Unfortunately, there are no general guidelines how to choose the value of this parameter and the correct value is rather case dependent. Since there was no better information available, in all the simulations a conservative value of 0.001 was used for the specularity coefficient to give some frictional effects at the walls. For the scalar equations no flux boundary conditions were used.

In the actual CFB, at the bottom of the riser there are 8 separate, equally spaced 0.013×0.013 m sized air injectors. Each of the injectors could have been modelled as a separate inlet, but again for simplicity the whole bottom boundary was specified as a single velocity inlet. The locations of the injectors were modelled with a velocity profile function shown in figure 11. In all the simulations it was assumed that the inlet gas velocity was constant and that the air flow was evenly distributed between the injectors. In reality though, the local pressure fluctuations of the bed just above an injector would affect the air flow through that nozzle. However, it would have made the simulations unnecessary complicated to take this phenomenon into account. Also, in this case the actual physical device was designed to have a relatively high pressure drop in the air distributor box under the injectors, so that the pressure fluctuations of the bed had less impact for the air flow through the inlets. In real industrial CFBs the air distribution system should probably be modelled more carefully since it is likely that any unnecessary pressure drops in the air distributors are eliminated to reduce power losses.

Since the recirculation path for the solids was excluded from the computations,

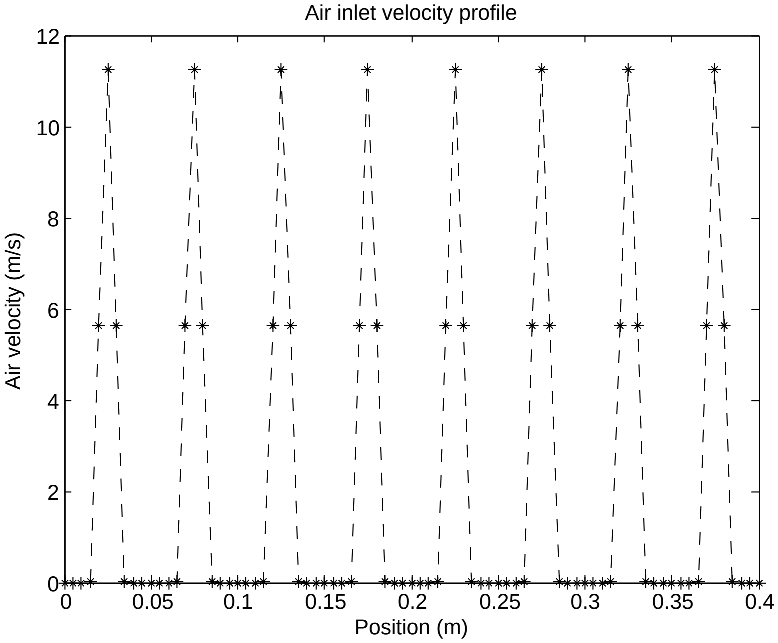

Figure 11: The gas velocity profile used in the boundary condition. In each simulation case the profile was scaled to match the desired superficial fluidization velocity.

special care had to be applied in order to maintain the proper amount of mass in the bed. In all cases it was assumed that the total mass inside the riser section should stay roughly constant. To ensure this, at each iteration the particle outflow through the outlet was calculated and the solids inflow through the recirculation channel was set to equal value. The same treatment was also applied for the different particle size classes, so that both the overall size distribution and the total mass remained unchanged during the simulation. Naturally, in the real device there is some amount of delay, before the escaped particles return to the riser and also the size distribution can change as some of the particles can remain in the loop seal. However, during steady state operation the inflow and outflow of the particles should be in balance at least on average and although the inflowing particles are not exactly the same as the outflowing particles, the flow rates would be similar.

6.4 Used models and solution strategies

When setting up granular multiphase flow simulation many different sub-models has to be chosen and quite a large amount of simulation parameters has to be specified. Fortunately, apart from the particle size distribution related options, most of the settings were common to all of the simulation cases. The settings used were generally either the default values used in FLUENT or otherwise they were found to give good results. In table 1 some of the most important simulation parameters are listed. The listing roughly follows the way the options are presented in the FLUENT and whenever possible, there is a corresponding reference to the theory section of this thesis. It should be noted, however, that the actual implementation of the various sub-models in FLUENT may differ from those presented in this thesis. Some information about the implementations can be found in the FLUENT manual.

The simulations were run as transient with a small time step of 0.5 ms and the initialization was done by patching a suitably sized homogeneous lump of solid material into the domain. At the beginning, the simulations were run until the initial transients had vanished and the system had reached a pseudo steady-state. After this, the calculation of the time averages was started. The time averaging was mostly straightforward, as the values of the variables were simply summed up and divided by the amount of time passed since the beginning of the averaging. However, since also the average values of some fluctuation terms were calculated, another averaging run was required. In the second run the reference value for calculating the fluctuations was taken from the first run.

When calculating time averages, it can be assumed that a longer sampling time would give more representative results. On the other hand, since the computations are very time consuming, in practice the sampling time has to be limited. As a compromise between the representativeness and required computation time, both averaging runs were about 60 seconds long. Within this time a reasonable representativeness could be expected.

Table 1: The models and parameter values used in the simulations.

Term	Model or value used
Collisional Viscosity	Gidaspow (26)
Granular Viscosity	Syamlal et al. (28)
Granular Bulk Viscosity	Lun et al. (29)
Frictional viscosity	Schaeffer (38)
Angle of internal friction	30°
Frictional Pressure	Based on the KTGF
Frictional Modulus	Derived
Friction Packing Limit, $\alpha_{\mathrm{s,min}}$	0.61
Granular Conductivity	Syamlal & O'Brien
Solid Pressure	Lun et al. (21)
Radial Distribution	Lun et al.
Elasticity Modulus	Derived
Packing Limit, $\alpha_{s,max}$	0.63
Drag Coefficient, K_{gs}	Gidaspow (45)
Restitution coefficient, e_{ss}	0.9

The simulation scheme was phase coupled SIMPLE and for volume fraction and momentum equations the QUICK-scheme was used as a spatial discretization. For the other equations first order upwind was used. Multiphase flow simulation tends to be quite prone to stability issues and for this reason a relatively low under-relaxation factors had to be used for momentum and volume fraction equations. Typically the factors were about 0.3 or even slightly less. The turbulence model used in the simulations was FLUENT's dispersed $k-\epsilon$ model and it's role was mainly to make the simulations more stable. The bed material in all cases consisted of spherical glass particles with a material density of 2480 kg/m^3.

Whenever a new simulation model is developed, it is a good idea to perform some general checks, to assert that there are no obvious problems with the implementation. For this reason, before the actual simulations were performed, a couple of simple tests were made. The non-linear equation solver was tested by solving some simple equation systems and comparing the results with those calculated with MATLAB. [65] The tests showed that the solver gave valid solutions. As a sanity check for the whole model, a case was run with three quadratures, where all the abscissas were identical. As expected, the system remained homogeneous with identical velocities for each size class and there were no changes in the relative amounts of the particles.

6.5 Experimental measurements

Because the experimental device had transparent walls, it was possible to measure the particle velocities and volume fractions with optical measurement methods. The particle velocities were determined by using Particle Image Velocimetry (PIV) and the particle volume fraction estimates were based on correlations for the light intensity of the image frames. The measurements were made at four different heights: 23, 40, 80 and 120 cm and at each height eight separate image sets were measured. A detailed description of an earlier, largely similar experimental setup with the same CFB device is given by Peltola et al. [66]

In PIV the fluid flow is seeded with small tracer particles and the flow is recorded with a high speed camera, which is capable of capturing pairs of images with only a very small time delay in between the frames. The images are divided into small interrogation areas and the interrogation area intensity fields are cross-correlated between the consequent frames in a pair. When the time delay between the two frames is known, the velocity of the particles can be calculated from the displacement of the correlation peaks. In this case, when the flow already contains solid particles, the addition of seed particles is unnecessary. The PIV can be applied directly to the solid particles and the local, average solid flow velocity can be obtained.

The solid volume fraction values were estimated based on light intensity of the recorded image frames. According to Grasa and Abanades, a simple logarithmic fitting function gives reasonably accurate estimates for the solid volume fraction. [67] The equation for the concentration of the particles C is

$$C = \frac{\log \frac{I}{I_{\min}}}{\log \frac{I_{\max}}{I_{\min}}}, \tag{130}$$

where I is the recorded light intensity and Imax and Imin are the light intensities from calibration reference images. When the equation (130) is applied to backlit images, the solid volume fraction can be estimated as

$$\alpha_{\mathrm{s}} = \alpha_{\mathrm{s,ref}} \frac{\log \frac{I_{\alpha_{\mathrm{s,ref}}}}{I}}{\log \frac{I_{\alpha_{\mathrm{s,0}}}}{I_{\alpha_{\mathrm{s,ref}}}}}, \tag{131}$$

where $\alpha_{\mathrm{s,ref}}$ is the particle volume fraction at the minimum intensity reference $I_{\alpha_{\mathrm{s,ref}}}$ and $I_{\alpha_{\mathrm{s,0}}}$ is the light intensity of the zero particle volume fraction. [66]

Although a single logarithmic fitting function can be used for the whole solid volume fraction range, it may be beneficial to do the correlation by parts. According to Peltola et al., the single fitting function tends to overestimate the volume fraction in dilute regions. Therefore if better accuracy is wanted, splitting the volume fraction range into two or more regions with separate calibration values would yield better estimates. Naturally, the cost of splitting the VOF range comes in the form of a more complicated calibration process.

6.6 Validity study of the local equilibrium approximation

The principal assumption of the mixture approach is the equilibrium approximation (109), where it is assumed that the differently sized particles have similar local acceleration. The validity of this kind of an approximation requires that the particles are rapidly accelerated to their terminal velocity with respect to the mixture, so that for example the length scale of the acceleration is smaller than the scale of the computational grid.

With the classic mixture formulation and in dilute flows a reasonable and simple way to approximate the time scale of acceleration is to calculate the relaxation time for a single particle in free fall. The free fall situation is represented by the drag equation (11) given in section 2.4. In simple cases, like for example for spherical particles, it is possible to solve the drag equation analytically and determine the relaxation time constant. For a spherical particle in low velocity Stokes flow the relaxation time is

$$t_{\mathrm{p}} = \frac{\rho_{\mathrm{p}} d_{\mathrm{p}}^2}{18\mu} \tag{132}$$

and for high velocity, constant C_{d} flows it is

$$t_{\mathrm{p}} = \frac{2\rho_{\mathrm{p}} d_{\mathrm{p}}}{3\rho_{\mathrm{g}} C_{\mathrm{d}} v_{\mathrm{t}}}, \tag{133}$$

where v_{t} is the terminal velocity. For a particle starting from rest it will take t_{p} to reach a velocity of $(1-1/e)v_{\mathrm{t}} \approx 0.63 v_{\mathrm{t}}$. [60]

With the solid mixture formulation and in dense flows it is more difficult to determine proper acceleration time scales, since besides the fluid-particle interaction, also the particle-particle forces have a big impact on the particle motion and the relaxation time greatly depends on the local solid concentration. For this reason the single particle in a free fall scenario does not accurately describe the typical situation and therefore also the relaxation times (132)-(133) represent poorly the actual time scales of acceleration. In order to get better estimates for the acceleration time scale, the solid concentration has to be included in the analysis.

Analytical derivation of a relaxation time for a particle in dense flow can be difficult, but it is possible to do numerical analysis in simplified cases. Let's consider a simple situation, where a mixture of particles is uniformly suspended by homogeneous gas flow in the vertical direction, so that the pressure and velocity gradients in the horizontal direction will vanish. Now if the pressure gradients in the vertical direction are also neglected as small, a momentum equation for some particle phase q can be written as

$$\frac{\partial v_q}{\partial t} = -g + \frac{1}{\alpha_q}\left(K_{\mathrm{g}q}(v_{\mathrm{g}} - v_q) + K'_{\mathrm{s}q}(v_{\mathrm{s}} - v_q)\right), \tag{134}$$

where the last term represents the average particle-particle drag between the particles q and the mixture. The equation (134) is now a simple differential equation and it can be easily solved numerically if the mixture volume fraction α_s and the velocities v_s and v_g are given.

As an example, let's consider some scenarios with realistic velocity and volume fraction values. The values used here are based on those obtained in the binary mixture simulation case, which is presented in the next section. In the bottom part of the riser, the average gas velocity is about 4 m/s, the solids velocity about 1.8 m/s, the solid volume fraction about 0.15 and the mixture diameter about 291 µm. Now let's assume, that there is a small amount of particles q, which are at rest at some point of time and assume that the gas and mixture variables have their average values. The equation (134) can then be solved giving the velocity v_q.

In figure 12 the velocity is presented for both 255 µm and 416 µm particles as a function of time. As can be seen from the figure, the acceleration is relatively fast and the relaxation time is around 0.30 ms for the larger particles, which is quite close but still below the flow time step of 0.5 ms. However, the acceleration length scale, which can be defined as $l_p = t_p u_t / e$, is approximately 0.20 mm for both cases and is clearly smaller than the 5 mm mesh size used in the simulations. Therefore in this case the equilibrium assumption seems to be reasonable.

On the other hand, in the upper part of the riser the solid volume fraction is lower and the equilibrium assumption is not so good. In figure 13 the velocity is shown for a case with $v_g = 2.5$ m/s, $v_s = 0.5$ m/s and $\alpha_s = 0.01$, which are typical values in the upper part of the riser. As seen in the figure, the acceleration lasts clearly longer, with the relaxation time being around $t_p = 25$ ms. Due to the low velocity, the relaxation length is still below the 5 mm, but with different velocities or in more dilute conditions the length scale will exceed the mesh size.

Even though it seems that the equilibrium approximation fails at the upper part of the riser, the overall error may not be very large. Firstly, the principal acceleration region for the particles is in the lower part of the riser, where the solid concentration is high and where the equilibrium is reached quickly. In the higher regions, where it takes longer to reach the equilibrium the particle velocity gradients are typically not so large, which makes the absolute errors smaller. Secondly, when calculating the relaxation times it was assumed that the particles q at rest would not affect the mixture properties. However, if the flow is dilute, then already a small number of particles can represent a substantial portion of the total mixture and thereby can affect the mixture velocity. Or if the amount of particles with a different velocity is very low, then it can be argued that the error in their velocity won't have a big impact on the overall flow behaviour.

Besides the two cases presented here, many variations of different gas and mixture velocities and solid volume fractions were studied. To summarize the findings, in dense regions with $\alpha_s > 0.1$ the equilibrium is usually reached quickly with most initial conditions and the equilibrium approach can be considered valid. With α_s 0.01 or less, the equilibrium assumption is not entirely valid and there can be problems if there are large accelerations.

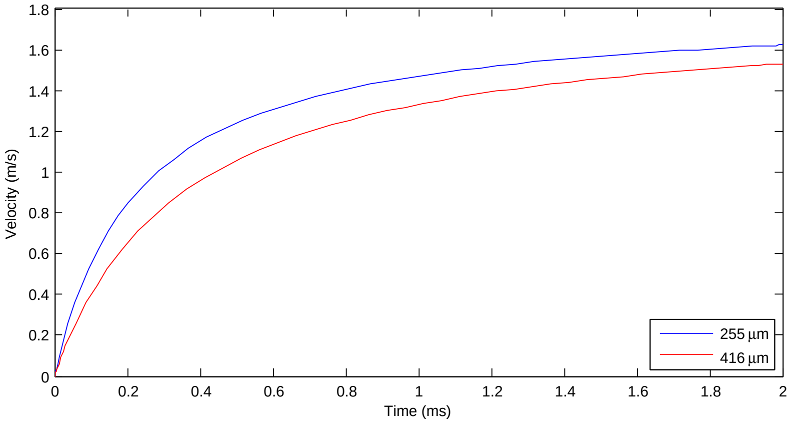

Figure 12: Relaxation velocities for 255 µm and 416 µm particles in dense flow.

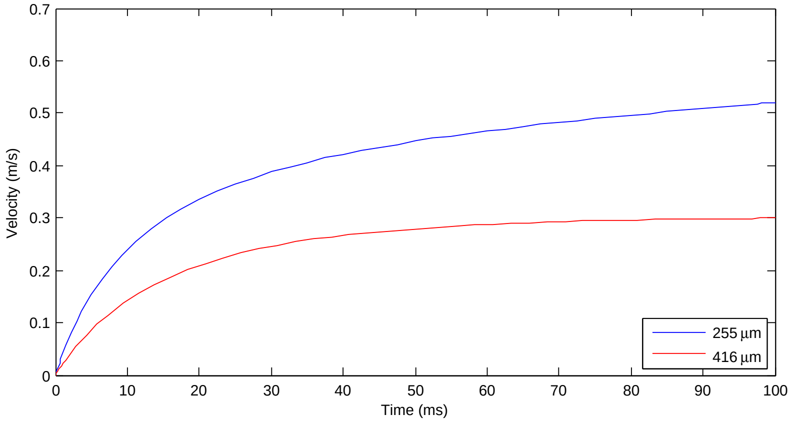

Figure 13: Relaxation velocities for 255 µm and 416 µm particles in dilute flow.

6.7 Case 1: Binary mixture

In the first simulation case the particles formed a binary mixture. In the experiments about 80 % of the particle mass consisted of small particles and the rest were clearly of larger size. The diameters of the particles were about 255 µm for the small ones and 416 µm for the larger ones and the mean diameter of the mixture was 279 µm. The exact parameters used in the case are presented in table 2. Since the distribution was binary, two quadratures were used in the simulation and the locations of the abscissas were manually chosen to match the sizes of the two types of particles. As a reference, a simulation run with the single, mean particle size was also performed.

Table 2: The modelling parameters for the simulation case 1.

Particle Distribution	78 % 255 µm 22 % 416 µm, SMD: 279 µm
Bed mass	2.85 kg
Fluidization velocity	2.25 m/s

Figure 14 shows a single snapshot from the simulation illustrating the simulated flow patterns. As seen in the figure, qualitatively the simulations have the similar complicated flow structure as the real flow has, with the particle clusters and strands clearly present. Just like in the real flow, the particle concentration is highest at the bottom and the flow gets more dilute along the height of the riser. On average the solid particles rise in the middle and come down as clusters near the walls. Naturally the used mesh size and the numerical diffusion filter out the very finest details of the flow, but overall the flow looks reasonably realistic.

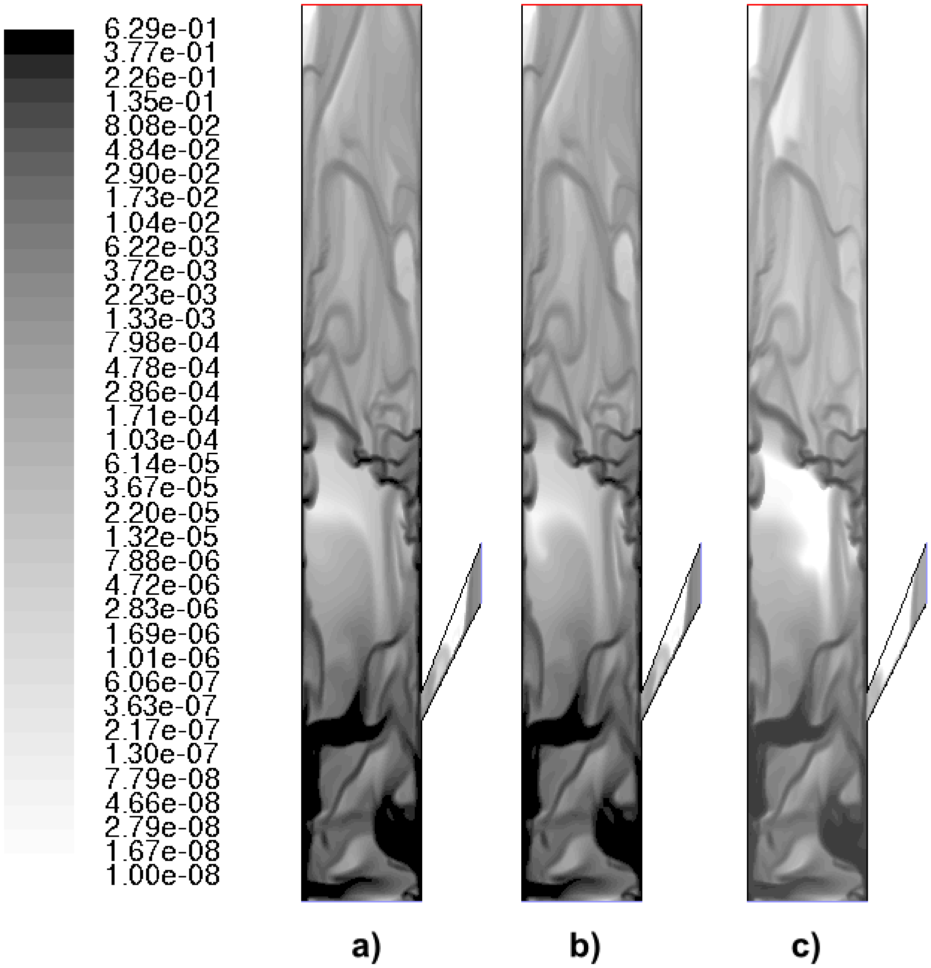

Figure 14: Instantaneous solid volume fractions: a) is for the mixture, b) for the 255 µm particles and c) for the 416 µm particles. The images have a logarithmic scale.

Since the flow in a fluidized bed is very unsteady, it is more useful to do quantitative comparison of the results through the averaged fields. The time averaged volume fractions of the both particle sizes, their sum and the volume fraction field from the reference simulation is shown in figure 15 and the vertical profiles of these fields is shown in figure 16. To make the comparison with the experimental results possible, also the horizontal volume fraction profiles from the four different measurement heights are shown in figure 17.

When looking at the volume fraction figures, it can be seen that the two different simulations give almost the same results, which is quite surprising. As there are no other reference data available, it is hard to say whether the similar results are caused by the mixture approach or if the effect of the PSD is simply small in this case. On one hand the implemented volume fraction correction procedure can cause the similarity of the results, because at each time step the total flux of the different size classes has to match the mixture flux. On the other hand, because the Sauter mean should give a representative average of the diameter for a particle mixture, the single diameter approximation should be relatively valid and the similarity of the results may well be reasonable.

When compared to the experimental results, the simulations are able to predict the volume fraction profiles with reasonable accuracy. However, in the dilute middle regions the experimental results show clearly larger volume fractions and near the walls there is a drop in the volume fractions, which is not visible in the simulations. The calibration process can be one of the reasons behind the observed differences. The experimental VOF values were calculated with a single logarithmic fitting function, which tends to give too high VOF estimates in the dilute regions as explained in the section 6.5. Also the amount of solid mass in the riser portion of the experimental device was not exactly known and the estimated solid mass value used in the simulations may differ from the actual value. This may be responsible for some of the differences in the results.

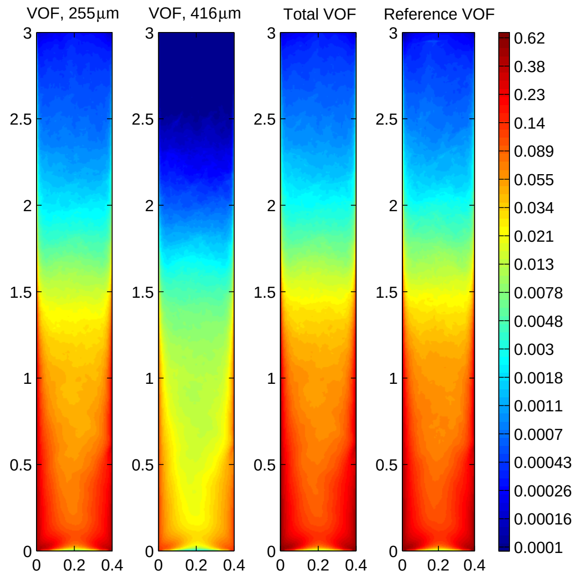

Figure 15: The average volume fraction fields on a logarithmic scale. The two images on the left represent the two particle size classes and the third image is their sum. The rightmost image is the volume fraction obtained from the reference simulation with one particle size.

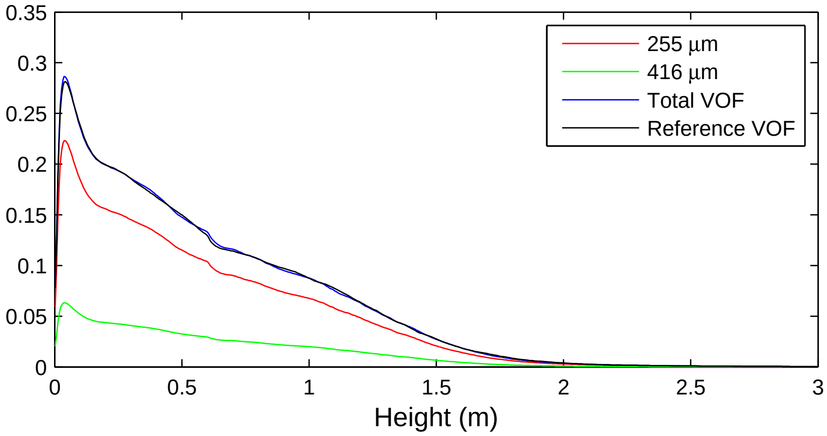

Figure 16: The mean volume fraction as a function of height.

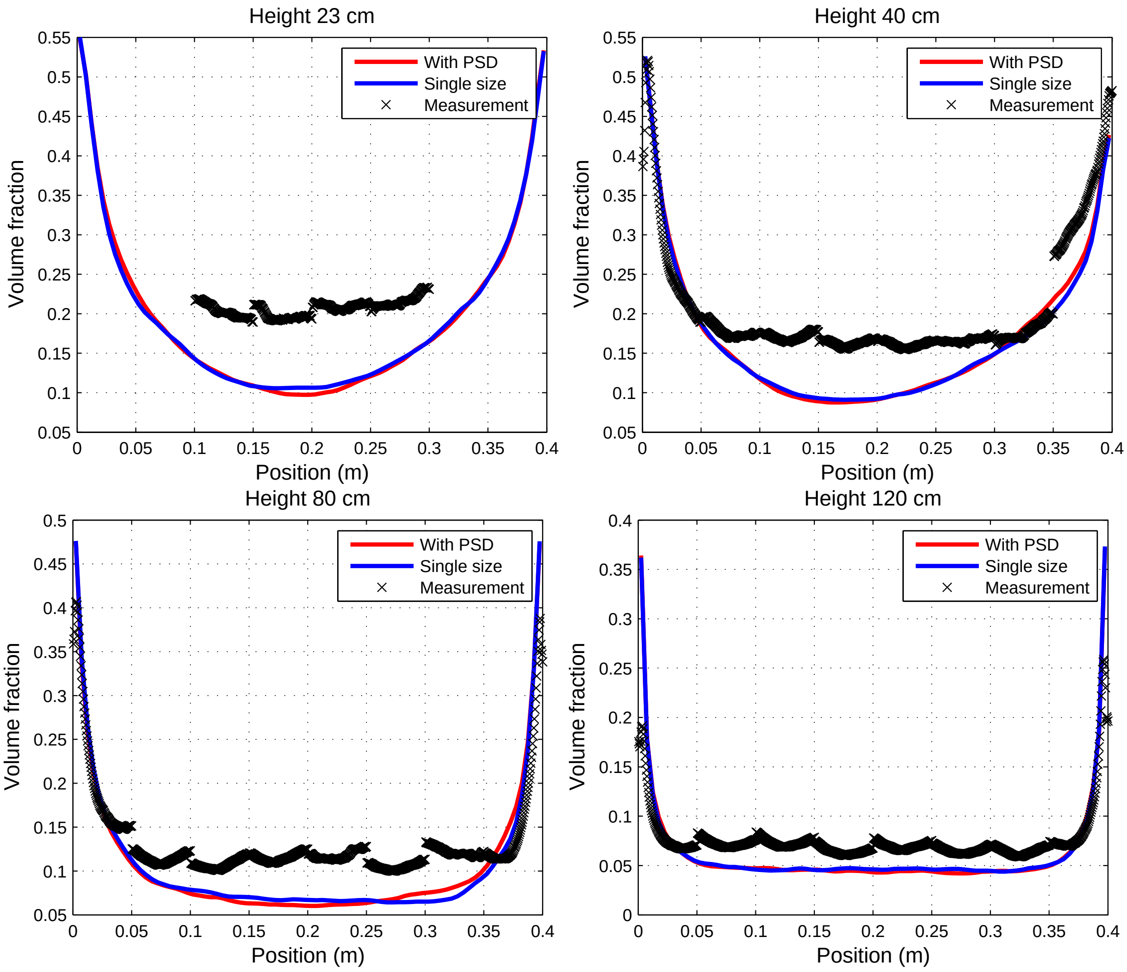

Figure 17: The mean volume fraction profiles at different heights.

Besides the average volume fraction fields, also the average particle velocity fields have much practical relevance. The Favre averaged solid velocity fields of the both simulations are shown in figure 18 and the velocity profiles together with the experimental results are shown in figure 19. Similarly as with the volume fraction fields, also here the differences between the two simulation cases are very small. In the upper part of the riser the differences are more noticeable, but this is most likely due to the relatively short averaging period. Higher up in the riser the time scales of the flow are longer than in the lower part and the averaging time period would have to be longer in order to get smooth and more representative results.

Also the experimental velocity profiles are reasonably well predicted in the simulations, although not as accurately as the volume fractions. The simulated velocity profiles show clearly higher velocities in the middle of the riser and near the walls. The differences in the velocities can be at least partly explained by the three dimensionality of the real device. In the real CFB, even though the device is thin, the particle clusters can pass each other in the depth direction without colliding. If some of the passing clusters are going up and the others are going down, the measured average velocity will be close to zero even though there is significant solid circulation in the riser. In 2D-simulation the same amount of circulation requires a greater velocity difference between the middle regions and the walls, as the clusters cannot simultaneously go up and down in the same region. What is also missing from the 2D-simulations is the effect of the front and back walls of the riser. It is likely that the friction caused by these walls slows down the solid particles in the real device.

The differences near the side walls can be explained by improper boundary conditions. As stated earlier, it is difficult to define the solid phase wall boundary condition correctly, because for example the proper value for the specularity coefficient is largely case dependant. In retrospect, in this case the chosen value of 0.001 seems to be a bit too small. Besides the specularity coefficient, also the restitution coefficient between the walls and the solid particles has to be given on the boundary. For same reason, the default value used by FLUENT is only 0.2, which is quite low given the physical meaning of the restitution coefficient. In the simulations this default value was assumed to be good enough and it was used without much consideration.

In order to check the effect of the boundary conditions, a separate test run with a single particle size and more reasonable boundary conditions was performed. In the new run the specularity coefficient was increased to 0.01 and the restitution coefficient was set to 0.9, which is the same as between the particles. The volume fraction and Favre averaged velocity profiles of this run are plotted in figures 20 and 21. As can be seen, the results are clearly improved near the walls and they show qualitatively the right kind of behaviour. Both the velocities and the solid volume fraction values are reduced near the walls. However, the effect of the boundaries does not reach very deep into the riser and therefore the velocity and volume fraction values are not affected much in the middle.

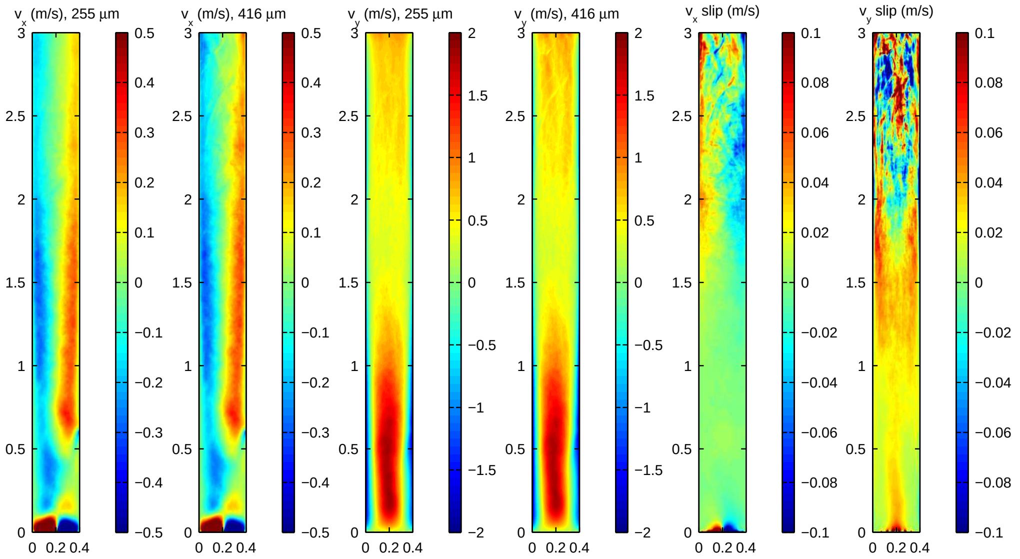

Figure 18: The Favre averaged velocity field components for both particle sizes. The two rightmost images show the difference between the two particle velocities.

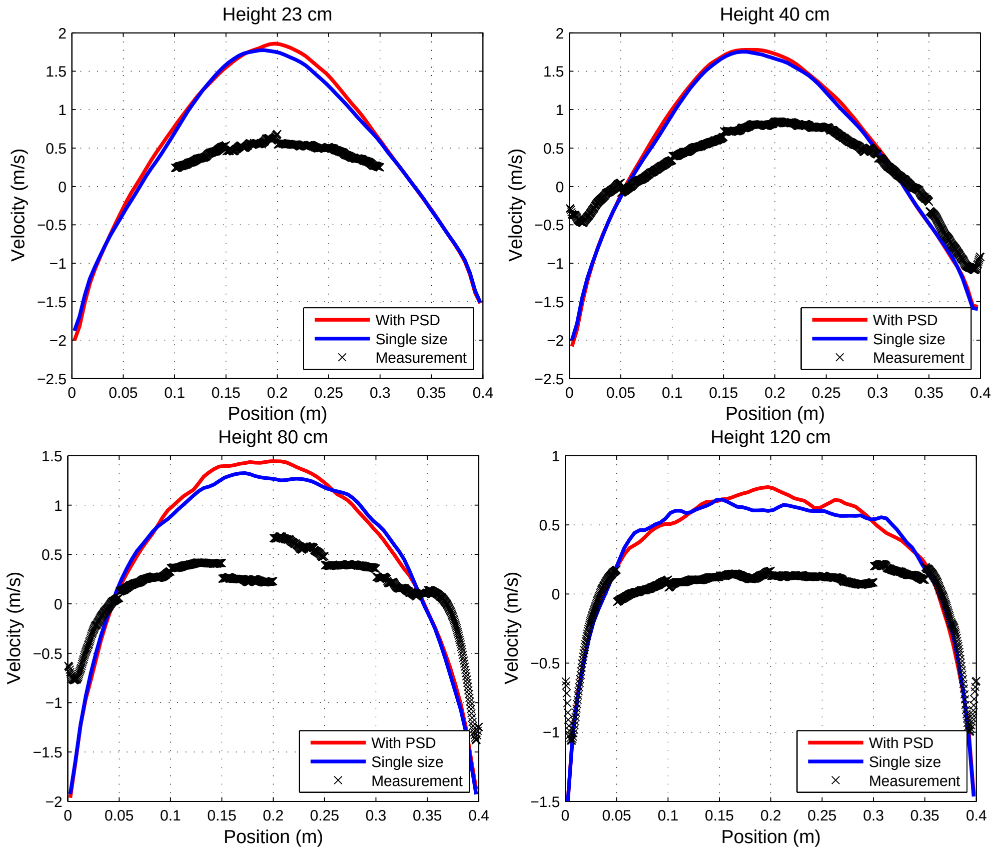

Figure 19: The Favre averaged profiles for vertical solid velocity.

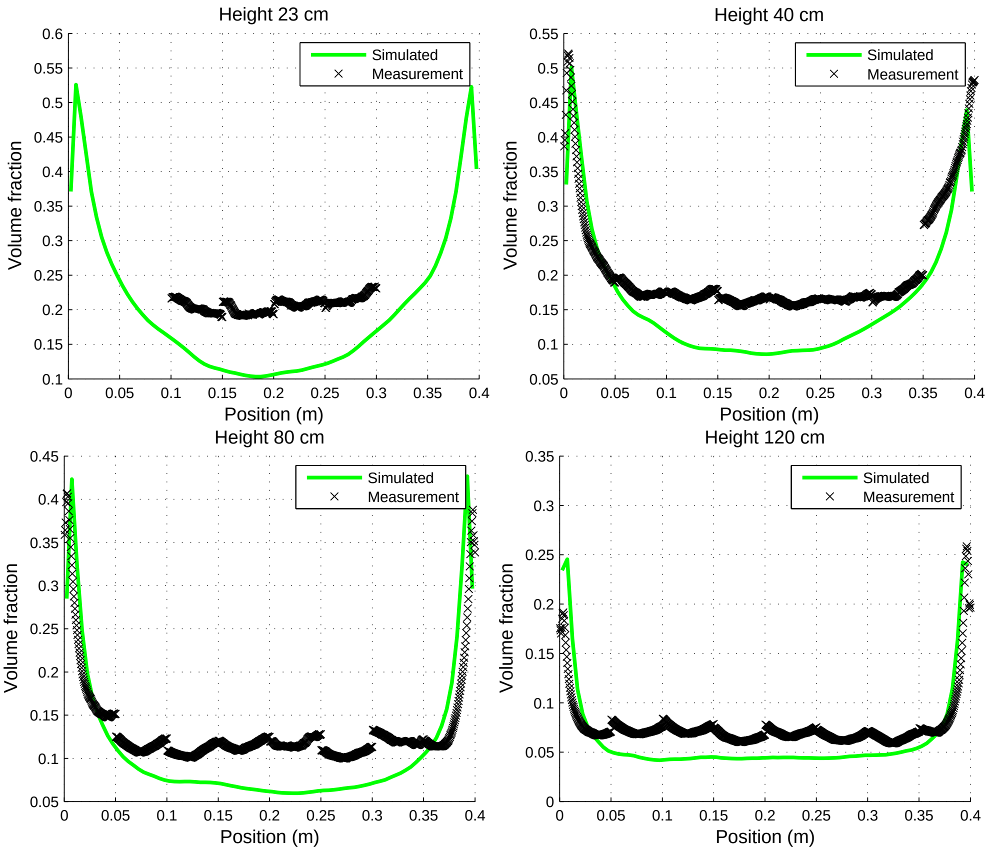

Figure 20: The mean volume fraction profiles at different heights, when different boundary conditions are used.

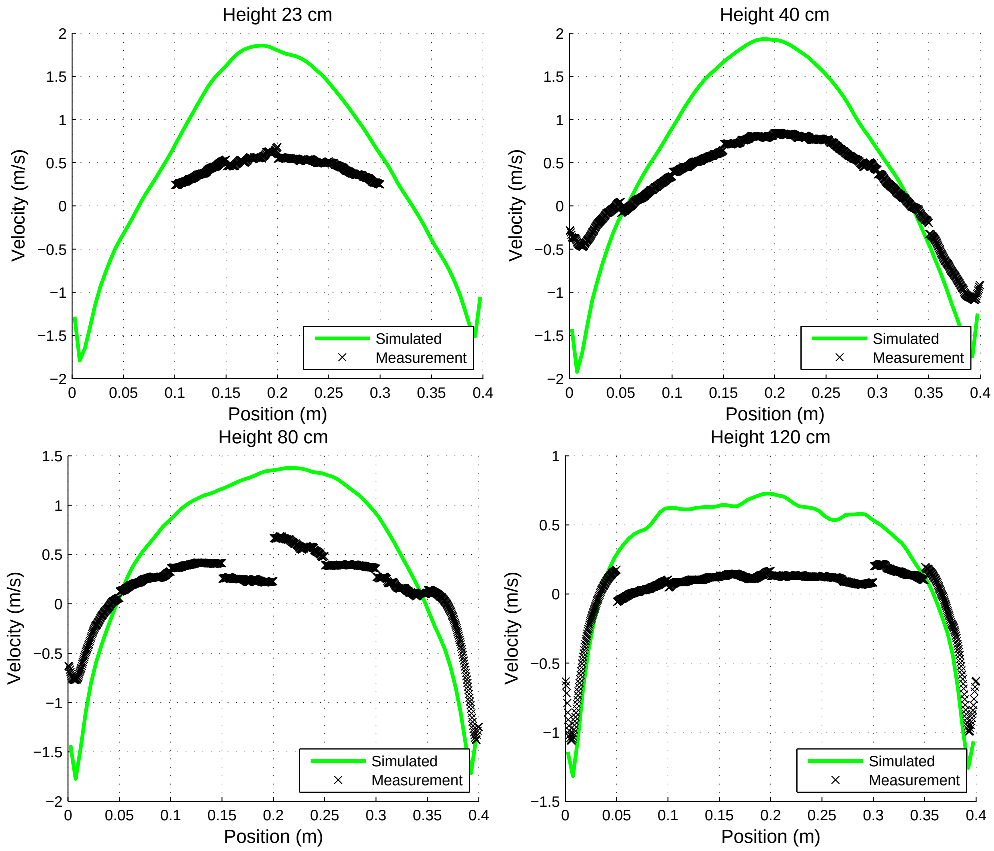

Figure 21: The Favre averaged profiles for vertical solid velocity, when different boundary conditions are used.

Although the mixture approach doesn't affect the volume fractions or the velocities much, it can still give some new insight to the flow behaviour. In the simulation with the PSD it is also possible to examine the average value of the particle diameter, which is just constant in the traditional simulations. The vertical profile of the mean diameter is plotted in figure 22 and qualitatively it looks reasonable. The diameter is largest at the bottom and it decreases along the height of the riser. Unfortunately, experimental measurements of the diameter were only made at samples taken from the bottom of the riser and from the loop seal, so the actual overall diameter field is unknown. However, comparisons can still be made with the samples and they show that the mixture approach is able to predict the solid diameter with good accuracy in the loop seal and at the bottom. In the experiments the particles at the bottom of the riser had a mean diameter of 306 µm and the samples from the loop seal had 264 µm. In the simulations the average diameter of the bottom two mesh rows was 304 µm and the mean diameter of the outflow through the outlet was 265.7 µm.

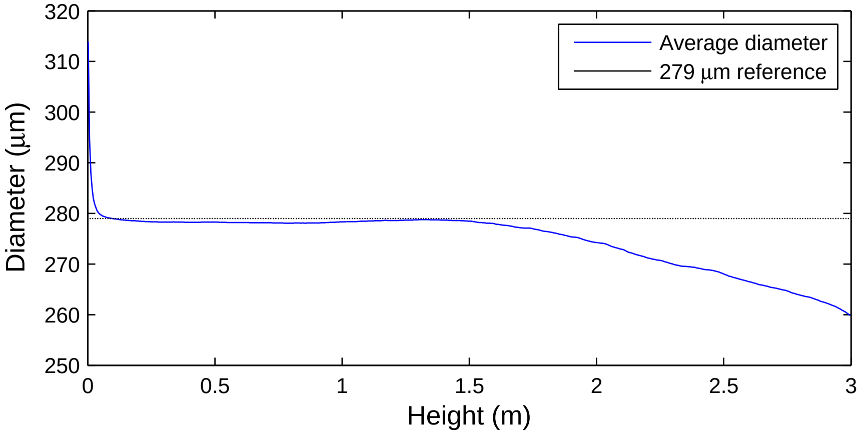

Figure 22: The mean particle diameter as a function of height.

6.8 Case 2: Wide size distribution

In the second experimental study, the particle size distribution was chosen to be roughly normally distributed to represent a more realistic situation inside a real CFB. Because the distribution was wider, three particle size classes were used in the simulation. The size distribution of the particle mixture is shown in figure 14 and also the locations and amounts of the initial volume fractions of the size classes are presented. The initial values for the size classes were calculated with the product-difference algorithm, which was implemented in MATLAB, to match the first six moments of the PSD. The other simulation properties are shown in table 3.

Table 3: The modelling parameters for the simulation case 2.

Particle size distribution	Wide, 75 µm-710 µm, SMD: 291 µm
Initial weights	α_1: 27.7 %, α_2: 66.9 %, α_3: 5.4 %
Size classes	L_1: 206 µm, L_2: 336 µm, L_3: 541 µm
Bed mass	2.86 kg
Fluidization velocity	2.25 m/s

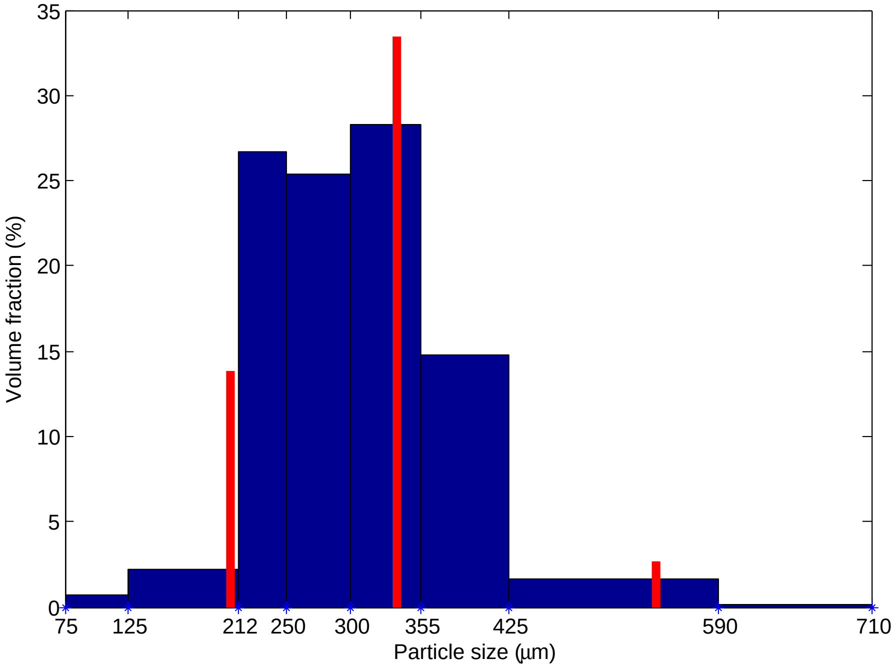

Figure 23: The particle size distribution in the case 2. The blue bars represent the actual distribution and the area of each bar is relative to the volume of particles in the corresponding interval. The red bars show to locations of the quadrature abscissas and their lengths are relative to their weights.

The results from the wide size distribution show similar trends as the results from the binary mixture. The volume fraction profiles are shown in figure 24 and the Favre averaged velocity profiles in figure 25. Again the mixture formulation gives practically the same results as the ones obtained from the reference simulations with the single, mean particle size. In general, the solid volume fraction profiles seem to be predicted a little bit better in this case and the velocity profiles have a similar accuracy as in the binary case. Again the quite large differences near the walls can be explained with the boundary conditions. The boundary condition experiments mentioned in the previous section were made only after these simulations and here the original values for specularity and restitution coefficients were used. Due to the relatively long computation times of the simulations, there was not enough time and resources to repeat the calculations with proper boundary conditions. If the proper boundary conditions were used, the results would presumably improve near the walls.

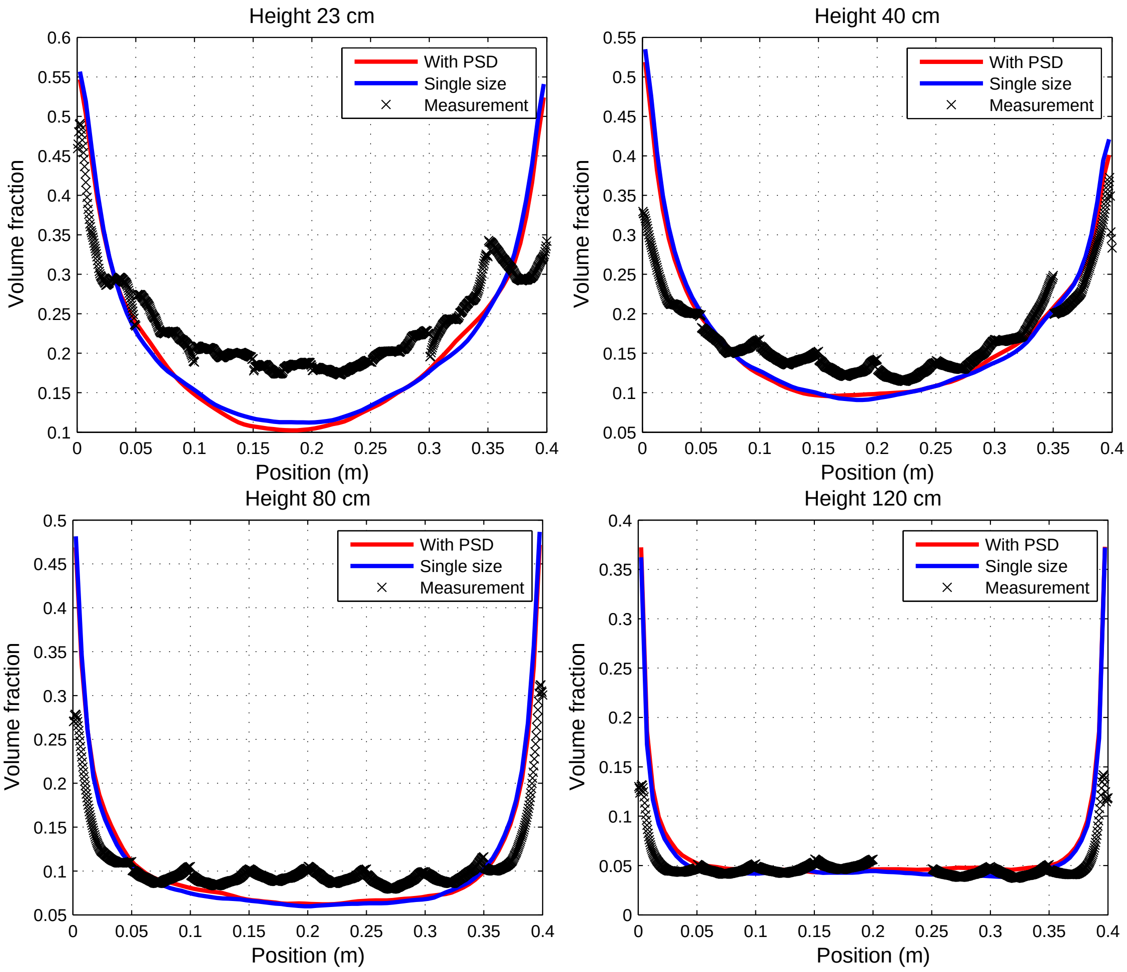

Figure 24: The mean volume fraction profiles at different heights.

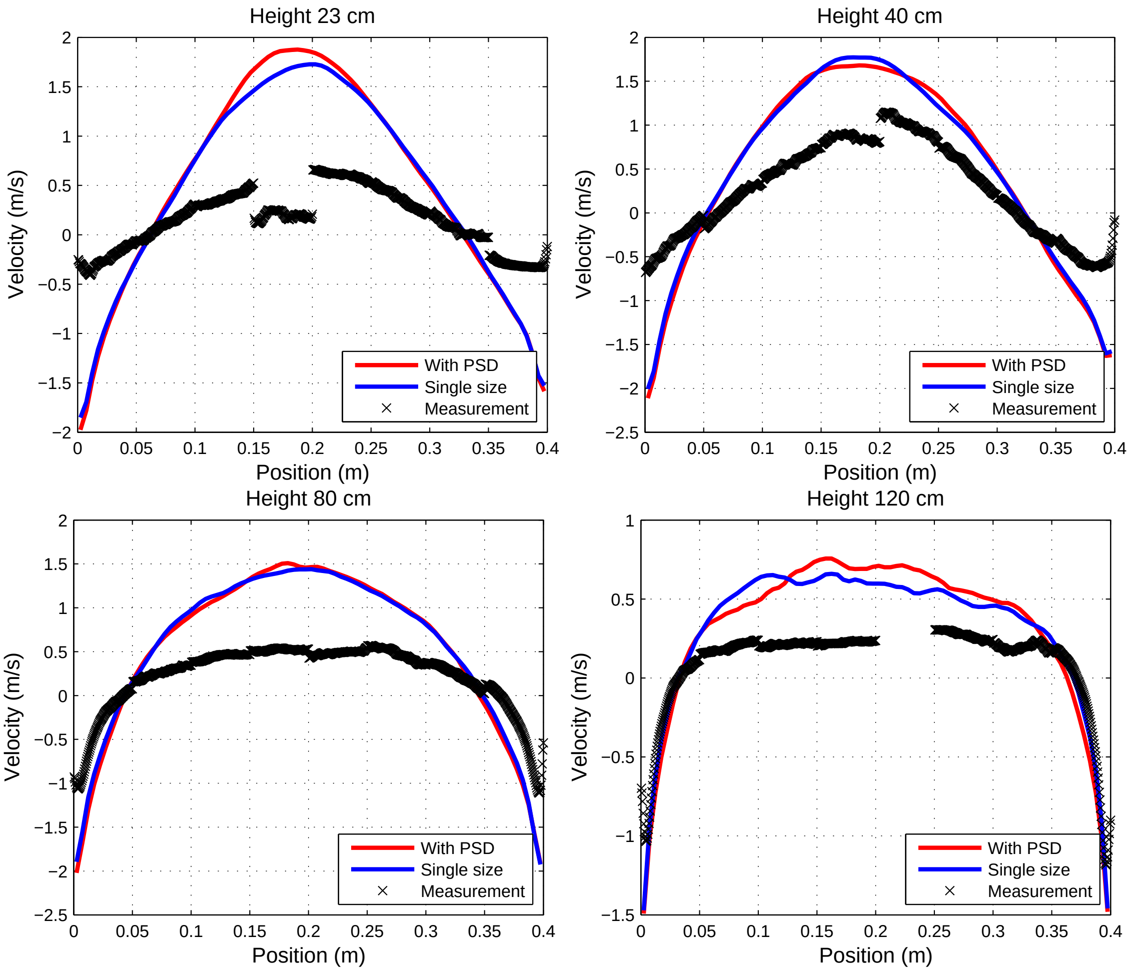

Figure 25: The Favre averaged profiles for vertical solid velocity.

The vertical profile of the mean diameter is plotted in figure 26 and it is also largely similar to the binary case. Again the experimental results for the particle diameter were available based on samples taken from the bottom of the riser and from the loop seal. According to the samples, the diameter at the bottom of the riser was 323 µm and in the loop seal it was 269 µm. The simulation results were 312 µm for the riser and 273 µm for the loop seal and therefore the diameters were again quite well predicted.

Both in this case and also in the binary case the simulated diameter is smaller than the experimentally obtained diameter at the bottom of the riser and larger at the outlet. This would suggest that in reality the slope of diameter profile should be a little bit more steep than what is predicted.

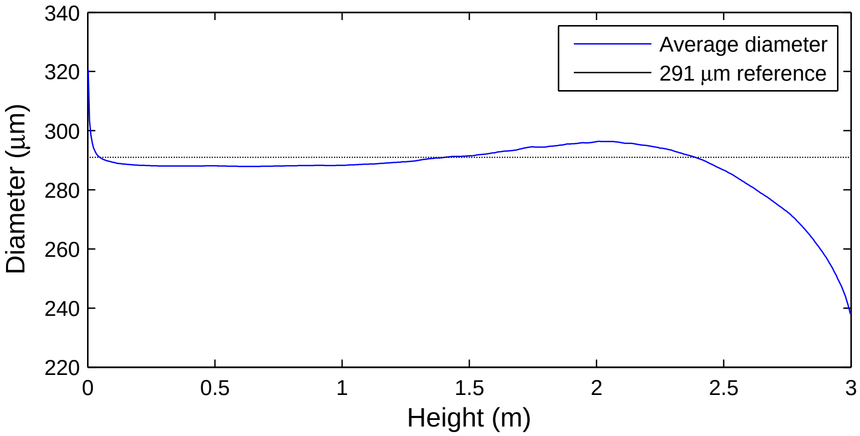

Figure 26: The mean particle diameter as a function of height.

6.9 The effect of the PSD

Because in the both simulation cases the mixture approach had only little effect on the volume fraction and solid velocity fields, the PSD does not seem to have a very large impact in these particular test cases. It is possible to estimate the effect of the size distribution by comparing the experimental results from the two different test cases. The experimental volume fraction and velocity profiles from the both cases are shown together in figures 27 and 28 and as can be seen, the differences between the two cases are indeed quite small. When taking into account that there is almost a 5 % difference in the mean diameters between the two cases, the role of distribution can be estimated to be very minor.

Given the similarity of the experimental results, the small effect of the mixture approach seems reasonable. However, for validation purposes it would be better, if the effect of the PSD was more clearly visible. Therefore, in order to better validate the mixture approach, further test cases with a larger impact from the PSD should be analysed.

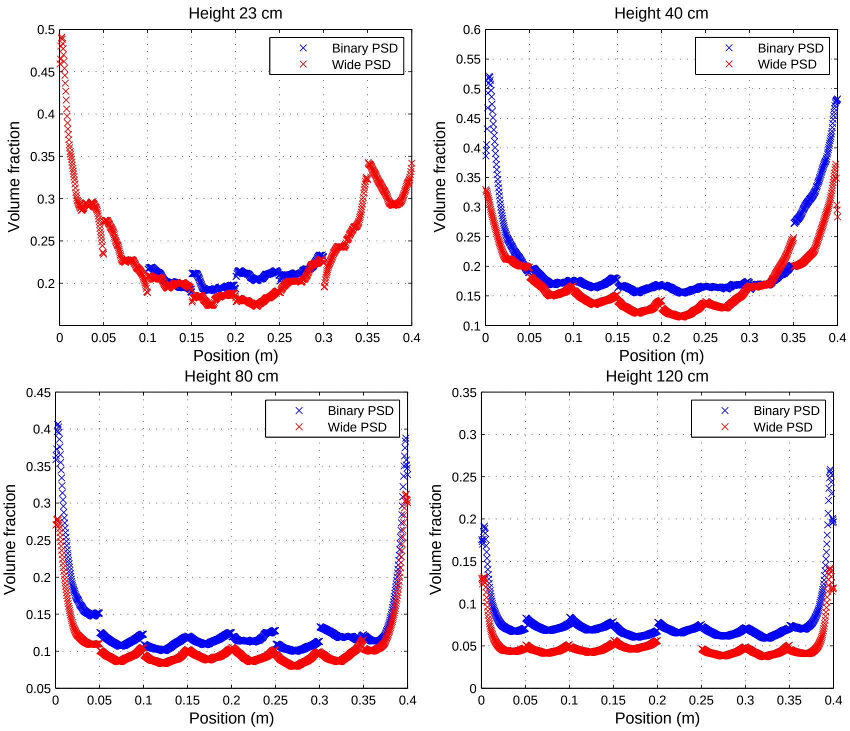

Figure 27: The experimental mean volume fraction profiles at different heights for the both test cases.

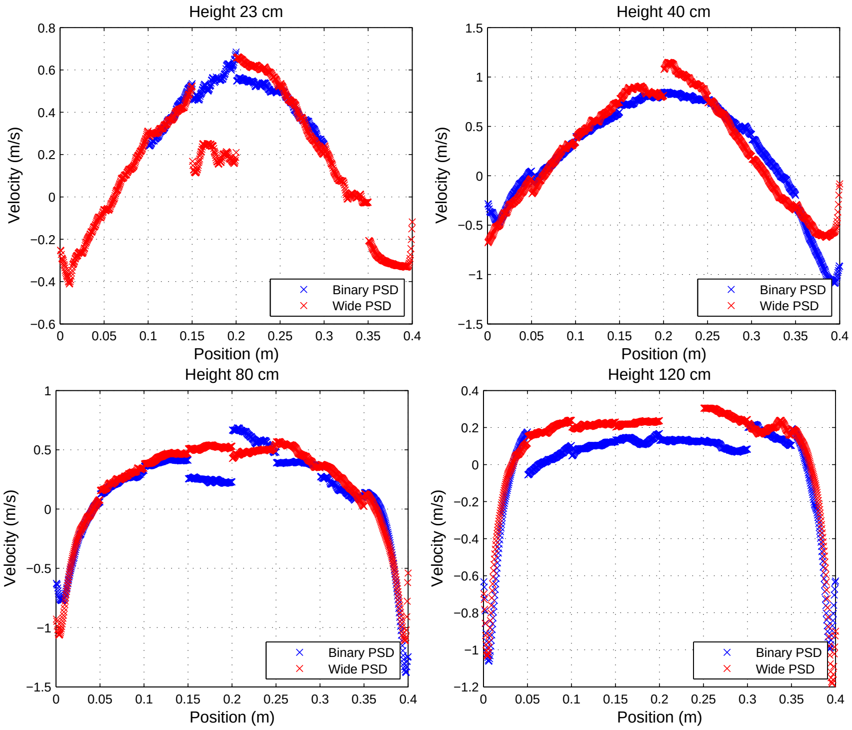

Figure 28: The experimental Favre averaged profiles for vertical solid velocity.

7 Conclusions

In this thesis the issue of including particle size distribution in CFD simulation of fluidized beds has been investigated. For the basic flow modelling the scope of the thesis is limited to transient Eulerian-Eulerian approach using the Kinetic Theory of Granular flow and all the simulations are performed with commercial CFD code FLUENT. When modelling the PSD, the particles are assumed to be inert and no size transfer mechanisms, such as attrition or fragmentation, are taken into account.

Traditionally only the mean diameter of the particles has been used in fludized bed flow simulations. However, both the increased computational resources and the recently developed new modelling approaches have made it viable to take also the shape of the PSD into account in the simulations. In this thesis several different modelling approaches are presented and their applicability to dense fluidized bed simulation is investigated. Of the various alternatives, the Direct Quadrature Method Of Moments is concluded to be most suitable for fluidized bed simulation. The DQMOM is based on the Quadrature Method Of Moments, in which only the moments of the particle size distribution are tracked and the integral terms are approximated with numerical quadratures.

Besides the Eulerian-Eulerian approaches, the Eulerian-Lagrangian approach using the dense DPM model was also briefly investigated. In the Eulerian-Lagrangian approach the particles are modelled as discrete objects with individual diameters and therefore it is straightforward and cheap to include the PSD in the simulations. The classical Lagrangian methods are too slow for practical fluidized bed modelling but the dense DPM approach should be comparable in speed to the Eulerian-Eulerian modelling making it an interesting alternative for modelling. Unfortunately in this thesis at least the current implementation of dense DPM in FLUENT was found out to be severely limited and unsuitable for fluidized bed modelling.

The existing DQMOM implementations require solving separate momentum equations for each different particle size class used in the simulations. This can increase the workload of the simulations substantially, which is not very desirable, as the transient simulations are very computationally demanding. In order to reduce the additional work, a simplified alternative approach using the mixture formulation was developed in this thesis.

In the mixture formulation the particle phase is modelled with a single momentum equation, which represents the average properties of all the different particles. The individual velocities for each size class are solved only approximately from algebraic force balance equations, which is faster than using separate momentum equations. The drawback of the mixture approach is that it requires making a number of approximations, which limit the applicability of the method.

The developed mixture formulation was implemented as a sub-model to FLUENT and the implementation was tested by simulating a couple of test cases with a lab-scaled CFB. The simulation results showed that in principle the mixture formulation is a feasible approach and can provide some additional insight to the simulated system. However, further studies with more comprehensive experimental and reference simulation results should be performed to conclusively assess the validity of the approach.

Besides doing validity studies, there are also many aspects which could be improved in the mixture formulation. First of all the volume fraction correction procedure should be improved to be more physically valid, if possible, as the current implementation is not very sophisticated. It could be also beneficial to try to reduce the amount of assumptions made during the derivation and implementation of the approach, because the assumptions can limit the applicability of the method.

More generally speaking the DQMOM and other moment methods should be also further developed, as the issues of moment corruption, ill-conditioning and problems with numerical diffusion can all hinder the usability of these methods in practical computations. Also the dense DPM method, event though it didn't work with FLUENT's implementation, shouldn't be left out of consideration. The validity of the method has been shown with other implementations and the theoretical background of the approach should be sound. With a good implementation it should be a very potential alternative for fluidized bed modelling.

In future it would be also interesting to see if it is possible to integrate some variation of the DQMOM, or the mixture approach presented here, into the framework of time averaged modelling. Time averaged modelling would allow fast simulation of industrial scale fluidized beds, which are very demanding for transient approaches due to large dimensions.

References

[1] Ansys inc., Ansys FLUENT. Computational Fluid Dynamics Software.

[2] Kunii, D. and Levenspiel, O. *Fluidization Engineering*. Butterworth-Heinemann, 2nd edition, 1991. ISBN 0-409-90233-0.

[3] Dasgupta, K., Joshi, J. B., and Banerjee, S. Fluidized bed synthesis of carbon nanotubes – a review. *Chemical Engineering Journal*, volume 171(3), 2011:pages 841 – 869. ISSN 1385-8947. doi:10.1016/j.cej.2011.05.038.

[4] Raiko, R., Saastamoinen, J., Hupa, M., and Kurki-Suonio, I., editors. *Poltto ja Palaminen*. Teknillistieteelliset akatemiat & International Flame Research Foundation - Suomen kansallinen osasto, 2002. ISBN 951-666-604-3. (In Finnnish).

[5] Kallio, S., Hermanson, A., Fagerudd, K., and Engblom, M. Experimental study of bubble interaction in a 2D bubbling fluidized bed cold model. Technical report, Åbo Akademi University, 2003.

[6] Geldart, D. Types of gas fluidization. *Powder Technology*, volume 7(5), 1973:pages 285 – 292. ISSN 0032-5910. doi:10.1016/0032-5910(73)80037-3.

[7] Ergun, S. Fluid flow through packed columns. *Chemical engineering progress, CEP*, volume 48, 1952:page 89.

[8] Yeoh, G. H. and Tu, J. *Computational Techniques for Multiphase Flows*. Butterworth-Heinemann, 2009. ISBN 978-0-08-046733-7.

[9] Fan, L.-S. and Zhu, C. *Principles of Gas-Solid Flows*. Cambridge Series in Chemical Engineering. Cambridge University Press, 2005. ISBN 978-0521021166. doi:10.2277/0521021162.

[10] Kleinstreuer, C. *Two-Phase Flow: Theory and Applications*. Taylor & Francis, 2003. ISBN 978-1591690009.

[11] Kallio, S., Seppälä, M., and Taivassalo, V. Time-averaged modelling of CFBs - Analysis of the terms in the momentum equation. Fluidization XIII - New paradigm in fluidization engineering. The 13th International Conference on Fluidization, Gyeong-ju, South Korea, 2010.

[12] Taivassalo, V., Kallio, S., and Peltola, J. On time-averaged CFD modeling of Circulating Fluidized beds. 12th International Conference on Multiphase flow in Industrial Plants, Ischia, Italy, 2011.

[13] Savage, S. B. and Jeffrey, D. J. The stress tensor in a granular flow at high shear rates. *Journal of Fluid Mechanics*, volume 110, 1981:pages 255–272. doi:10.1017/S0022112081000736.

[14] Jenkins, J. T. and Savage, S. B. A theory for the rapid flow of identical, smooth, nearly elastic, spherical particles. *Journal of Fluid Mechanics*, volume 130, 1983:pages 187–202. doi:10.1017/S0022112083001044.

[15] Ding, J. and Gidaspow, D. A bubbling fluidization model using kinetic theory of granular flow. *AIChE Journal*, volume 36(4), 1990:pages 523–538. ISSN 1547-5905. doi:10.1002/aic.690360404.

[16] Gidaspow, D. *Multiphase Flow and Fluidization - Continuum and Kinetic Theory Descriptions*. Academic Press, 1994. ISBN 0-12-282470-9.

[17] Hiltunen, K., Jäsberg, A., Kallio, S., Karema, H., Kataja, M., Koponen, A., Manninen, M., and Taivassalo, V. *Multiphase Flow Dynamics: Theory and Numerics*. VTT Publications : 722. VTT, 2009. ISBN 978-951-38-7365-3.

[18] Lun, C. K. K., Savage, S. B., Jeffrey, D. J., and Chepurniy, N. Kinetic theories for granular flow: inelastic particles in couette flow and slightly inelastic particles in a general flowfield. *Journal of Fluid Mechanics*, volume 140, 1984:pages 223–256. doi:10.1017/S0022112084000586.

[19] Syamlal, M., Rogers, W., and O'Brien, T. J. *MFIX Documentation: Volume 1, Theory Guide*. National Technical Information Service, Springfield, VA, 1993. DOE/METC-9411004, NTIS/DE9400087.

[20] Ahmadi, G. and Ma, D. A thermodynamical formulation for dispersed multiphase turbulent flows—1: Basic theory. *International Journal of Multiphase Flow*, volume 16(2), 1990:pages 323 – 340. ISSN 0301-9322. doi:10.1016/0301-9322(90)90062-N.

[21] Ogawa, S., Umemura, A., and Oshima, N. On the Equations of Fully Fluidized Granular Materials. *Journal of Applied Mathematics and Physics (ZAMP)*, volume 31, 1980:pages 483–493. ISSN 0044-2275. doi:10.1007/BF01590859.

[22] Johnson, P. C. and Jackson, R. Frictional-collisional constitutive relations for granular materials, with application to plane shearing. *Journal of Fluid Mechanics*, volume 176, 1987:pages 67–93. doi:10.1017/S0022112087000570.

[23] Schaeffer, D. G. Instability in the evolution equations describing incompressible granular flow. *Journal of Differential Equations*, volume 66(1), 1987:pages 19 – 50. ISSN 0022-0396. doi:10.1016/0022-0396(87)90038-6.

[24] Wen, C. and Yu, Y. Mechanics of fluidization. *Chem. Eng. Prog. Symp. Ser.*, volume 62(62), 1966:pages 100–111.

[25] Leboreiro, J., Joseph, G. G., and Hrenya, C. M. Revisiting the standard drag law for bubbling, gas-fluidized beds. *Powder Technology*, volume 183(3), 2008:pages 385 – 400. ISSN 0032-5910. doi:10.1016/j.powtec.2008.01.008.

[26] Deen, N. G., Van Sint Annaland, M., Van der Hoef, M. A., and Kuipers, J. A. M. Review of discrete particle modeling of fluidized beds. *Chemical Engineering*

Science, volume 62(1-2), 2007:pages 28–44. ISSN 0009-2509. doi:10.1016/j.ces.2006.08.014.

[27] Andrews, M. and O'Rourke, P. The multiphase particle-in-cell (MP-PIC) method for dense particulate flows. *International Journal of Multiphase Flow*, volume 22(2), 1996:pages 379 – 402. ISSN 0301-9322. doi:10.1016/0301-9322(95)00072-0.

[28] Snider, D. M., O'Rourke, P. J., and Andrews, M. J. An Incompressible Two-Dimensional Multiphase Particle-in-Cell Model for Dense Particle Flows. LA-13280-MS, Los Alamos National Laboratories, Los Alamos, NM, 1997.

[29] Snider, D. M. An Incompressible Three-Dimensional Multiphase Particle-in-Cell Model for Dense Particle Flows. *Journal of Computational Physics*, volume 170(2), 2001:pages 523 – 549. ISSN 0021-9991. doi:10.1006/jcph.2001.6747.

[30] Snider, D. M. Three fundamental granular flow experiments and CPFD predictions. *Powder Technology*, volume 176, 2007:pages 36 – 46. ISSN 0032-5910. doi:10.1016/j.powtec.2007.01.032.

[31] Yu, Y., Zhou, L., Wang, B., and Cai, F. A USM-Θ two-phase turbulence model for simulating dense gas-particle flows. *Acta Mechanica Sinica*, volume 21, 2005:pages 228–234. ISSN 0567-7718. doi:10.1007/s10409-005-0037-7.

[32] Zeng, Z. and Zhou, L. A two-scale second-order moment particle turbulence model and simulation of dense gas-particle flows in a riser. *Powder Technology*, volume 162(1), 2006:pages 27 – 32. ISSN 0032-5910. doi:10.1016/j.powtec.2005.10.011.

[33] Zhou, L. Second-order moment modeling of dispersed two-phase turbulence-Part 2-USM-Θ two-phase turbulence model and USM-SGS two-phase stress model. *SCIENCE CHINA Physics, Mechanics & Astronomy*, volume 54, 2011:pages 1296–1303. ISSN 1674-7348. doi:10.1007/s11433-011-4322-0.

[34] Versteeg, H. K. and Malalasekera, W. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Prentice Hall, 2nd edition, 2007. ISBN 978-0-13-127498-3.

[35] Ferziger, J. H. and Peric, M. *Computational Methods for Fluid Dynamics*. Springer, 3rd edition, 2002. ISBN 978-3-540-42074-3.

[36] Ramkrishna, D. *Population Balances: Theory and Applications to Particulate Systems in Engineering*. Academic Press, London, 2000. ISBN 0-12-576970-9.

[37] Gonzales, O. and Stuart, A. M. *A First Course in Continuum Mechanics*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2008. ISBN 978-0-521-71424-2.

[38] Lo, S. Application of population balance to CFD modeling of bubbly flow via the MUSIG model. Technical report, AEAT-1096, AEA Technology plc, 1996.

[39] Krepper, E., Lucas, D., and Prasser, H.-M. On the modelling of bubbly flow in vertical pipes. *Nuclear Engineering and Design*, volume 235(5), 2005:pages 597–611. ISSN 0029-5493. doi:10.1016/j.nucengdes.2004.09.006.

[40] Shi, J.-M., Zwart, P., Frank, T., Rohde, U., and Prasser, H.-M. Development of a multiple velocity multiple size group model for poly-dispersed multiphase flows, 2004. Annual report of Institute of Safety Research. Germany: Forschungszentrum Rossendorf.

[41] Frank, T., Zwartz, P. J., Shi, J.-M., Krepper, E., Lucas, D., and Rohde, U. Inhomogeneous MUSIG model-A population balance approach for polydispersed bubbly flows. In *proceedings of International Conference: "Nuclear Energy for New Europe"*. Bled, Slovenia, September 5-8, 2005.

[42] Krepper, E., Lucas, D., Frank, T., Prasser, H.-M., and Zwart, P. J. The inhomogeneous MUSIG model for the simulation of polydispersed flows. *Nuclear Engineering and Design*, volume 238, 2008:pages 1690–1702. doi: 10.1016/j.nucengdes.2008.01.004.

[43] Lucas, D., Frank, T., Lifante, C., Zwart, P., and Burns, A. Extension of the inhomogeneous MUSIG model for bubble condensation. *Nuclear Engineering and Design*, volume 241(11), 2011:pages 4359 – 4367. ISSN 0029-5493. doi: 10.1016/j.nucengdes.2010.10.039.

[44] Hulburt, H. M. and Katz, S. Some problems in particle technology: A statistical mechanical formulation. *Chemical Engineering Science*, volume 19(8), 1964:pages 555 – 574. ISSN 0009-2509. doi:10.1016/0009-2509(64)85047-8.

[45] McGraw, R. Description of aerosol dynamics by the quadrature method of moments. *Aerosol Science and Technology*, volume 27(2), 1997:pages 255–265. doi:10.1080/02786829708965471.

[46] Gordon, R. G. Error Bounds in Equilibrium Statistical Mechanics. *Journal of Mathematical Physics*, volume 9(5), 1968:pages 655–663. doi:10.1063/1.1664624.

[47] Marchisio, D. L., Pikturna, J. T., Fox, R. O., Vigil, R. D., and Barresi, A. A. Quadrature method of moments for population-balance equations. *AIChE Journal*, volume 49(5), 2003:pages 1266–1276. ISSN 1547-5905. doi:10.1002/aic.690490517.

[48] Marchisio, D. L., Vigil, R., and Fox, R. O. Quadrature method of moments for aggregation–breakage processes. *Journal of Colloid and Interface Science*, volume 258(2), 2003:pages 322 – 334. ISSN 0021-9797. doi: 10.1016/S0021-9797(02)00054-1.

[49] Marchisio, D. L., Vigil, R. D., and Fox, R. O. Implementation of the quadrature method of moments in CFD codes for aggregation–breakage problems. *Chemical Engineering Science*, volume 58(15), 2003:pages 3337 – 3351. ISSN 0009-2509. doi:10.1016/S0009-2509(03)00211-2.

[50] Sanyal, J., Marchisio, D. L., Fox, R. O., and Dhanasekharan, K. On the Comparison between Population Balance Models for CFD Simulation of Bubble Columns. *Industrial & Engineering Chemistry Research*, volume 44(14), 2005:pages 5063–5072. doi:10.1021/ie049555j.

[51] Silva, L. F. L. R., Rodrigues, R., Mitre, J., and Lage, P. Comparison of the accuracy and performance of quadrature-based methods for population balance problems with simultaneous breakage and aggregation. *Computers & Chemical Engineering*, volume 34(3), 2010:pages 286 – 297. ISSN 0098-1354. doi:10.1016/j.compchemeng.2009.11.005.

[52] Selma, B., Bannari, R., and Proulx, P. Simulation of bubbly flows: Comparison between direct quadrature method of moments (DQMOM) and method of classes (CM). *Chemical Engineering Science*, volume 65(6), 2010:pages 1925 – 1941. ISSN 0009-2509. doi:10.1016/j.ces.2009.11.018.

[53] Marchisio, D. L. and Fox, R. O. Solution of population balance equations using the direct quadrature method of moments. *Journal of Aerosol Science*, volume 36(1), 2005:pages 43 – 73. ISSN 0021-8502. doi:10.1016/j.jaerosci.2004.07.009.

[54] Fan, R., Marchisio, D. L., and Fox, R. O. Application of the direct quadrature method of moments to polydisperse gas–solid fluidized beds. *Powder Technology*, volume 139(1), 2004:pages 7 – 20. ISSN 0032-5910. doi:10.1016/j.powtec.2003.10.005.

[55] Fan, R. and Fox, R. O. Segregation in polydisperse fluidized beds: Validation of a multi-fluid model. *Chemical Engineering Science*, volume 63(1), 2008:pages 272 – 285. ISSN 0009-2509. doi:10.1016/j.ces.2007.09.038.

[56] Grosch, R., Briesen, H., Marquardt, W., and Wulkow, M. Generalization and Numerical Investigation of QMOM. *AIChE Journal*, volume 53(1), 2007:pages 207–227. ISSN 1547-5905. doi:10.1002/aic.11041.

[57] Wright, D. L., Jr. Numerical advection of moments of the particle size distribution in Eulerian models. *Journal of Aerosol Science*, volume 38(3), 2007:pages 352 – 369. ISSN 0021-8502. doi:10.1016/j.jaerosci.2006.11.011.

[58] Mazzei, L., Marchisio, D. L., and Lettieri, P. New Quadrature-Based Moment Method for the Mixing of Inert Polydisperse Fluidized Powders in Commercial CFD Codes. *AIChE Journal*, 2012. ISSN 1547-5905. doi:10.1002/aic.13714. Online early view.

[59] Mazzei, L., Marchisio, D. L., and Lettieri, P. Direct Quadrature Method of Moments for the Mixing of Inert Polydisperse Fluidized Powders and the Role of Numerical Diffusion. *Industrial & Engineering Chemistry Research*, volume 49(11), 2010:pages 5141–5152. doi:10.1021/ie901116y.

[60] Manninen, M., Taivassalo, V., and Kallio, S. *On the mixture model for multiphase flow*. VTT Publications 288, Technical Research Centre of Finland, 1996. ISBN 951-38-4946-5.

[61] Syamlal, M. The Particle-Particle Drag Term in a Multiparticle Model of Fluidization. Technical report, National Technical Information Service, Springfield, VA, 1987. DOE/MC/21353-2373, NTIS/DE87006500.

[62] Guldén, M. *Pilotmodell av en cirkulerande fluidiserad bädd*. (In Swedish), Åbo Akademi University, Turku, Finland, 2008.

[63] Kallio, S., Airaksinen, J., Guldén, M., Hermanson, A., Peltola, J., Ritvanen, J., Seppälä, M., Shah, S., and Taivassalo, V. Experimental and numerical study of hydrodynamics in a circulating fluidized bed. In *Proceedings of Finnish-Swedish Flame Days*. Naantali, Finland, 2009.

[64] Kallio, S., Guldén, M., and Hermanson, A. Experimental Study and CFD Simulation of a 2D Circulating Fluidized Bed. In G. Yue, H. Zhang, C. Zhao, and Z. Luo, editors, *Proceedings of the 20th International Conference on Fluidized Bed Combustion*. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-02682-9, pages 799–804. doi:10.1007/978-3-642-02682-9_123.

[65] The MathWorks Inc. MATLAB version 7.9.0 (r2009b), 2009. A numerical computing environment.

[66] Peltola, J., Kallio, S., Honkanen, M., and Saarenrinne, P. Image based measurement of particle phase reynolds stresses in a laboratory scale circulating fluidized bed. In *7th International Conference on Multiphase Flow ICMF2010, May 30 - June 4, 2010, Tampa, Florida 9 p.*

[67] Grasa, G. and Abanades, J. A calibration procedure to obtain solid concentrations from digital images of bulk powders. *Powder Technology*, volume 114(1–3), 2001:pages 125 – 128. ISSN 0032-5910. doi:10.1016/S0032-5910(00)00262-X.