International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

CFD DEM Study of Gas Solid Fluidized Bed for Non Spherical Particles

Shubham

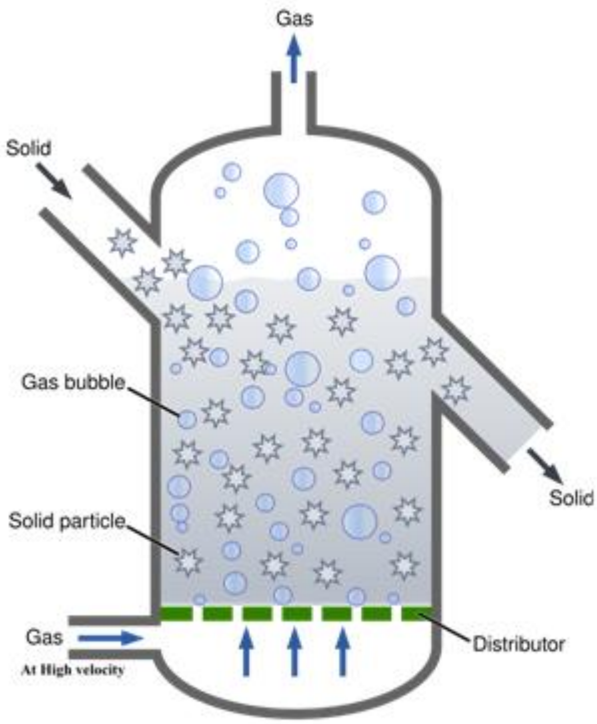
Department of Chemical Engineering Indian Institute of Technology Delhi New Delhi – 110016, India

Abstract: ***This project reviews the study of simulation results performed for non spherical particles inside a Fluidized bed at different velocities, and we will be using the MFIX - SQP model to simulate this case. Mfix is an open source CFD software package developed by NETL for simulating multiphase flows in scientific and engineering applications. But before going deep into simulations of non - spherical particles, we will be understanding the fundamentals of Fluidized Bed and its principles and the models which are being used and also we will compare different models as wells, the difference which comes between spherical and non - spherical particle modeling. In this project, we will be analyzing the particles behavior and at different gas velocity and time frame.***

Keywords: CFD, DEM, Gas Solid Fluidized Bed

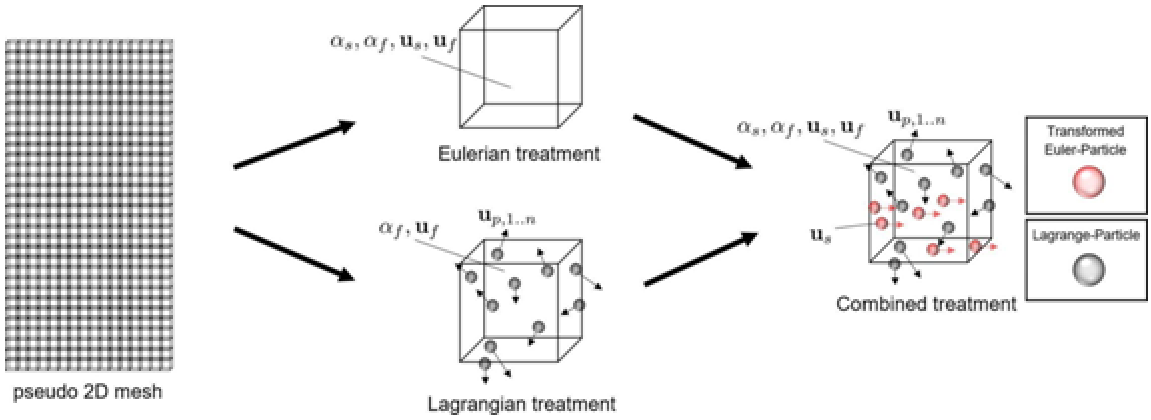
1. Introduction

Fluidized beds are used widely in the chemical and process industries for processes like drying, combustion gasification, and catalytic reactions. The fluidized bed reactors offer numerous advantages over other reactors, including better heat and mass transfer, better mixing and temperature control. But these reactors are complex to understand due to interaction between solid and gas particles, which makes it difficult to predict. In a fluidized bed, a bed of small solid particles are suspended and kept in motion by an upward flow of gas or liquid. In order to analyze the behavior of particles in fluidized beds, a thorough understanding of transport phenomena in these systems is required.



Fluidized bed Reactor

There is a vast amount of literature in existence on experimental study of fluidized beds. They often use pseudo 2D fluidized beds to study the fluidization behavior by video techniques, because 3D beds are not accessible visually. To solve these limitations in practical experiments, Computer models gained attention from the early 1990s. The reason behind is because one can look inside the bed without even disturbing the flow of particles. The use of simulations, specifically the discrete element model (DEM), has enabled us to measure the properties like particle and gas velocities as well as porosity which were difficult to find through direct experimentation. In simulations, we also have the advantage of testing several design options and operating conditions with much ease. But the creation of a reliable model which can be leveraged to large - scale gas - solid is still challenging because of lack of fundamental understanding of dense gas - solid contractors, especially the phenomena which is related to effective particle - particle, particle - gas interactions, and particle - wall interactions.



Eulerian Vs Lagrangian treatment from a 2D mesh and its combined treatment

Also the multiscale nature of this phenomena make it more complex because the interactions take place at molecular level which is a much smaller scale than the flow structure, which can be of meters in size, resulting in large separation of scale. To find a solution, discrete (Lagrangian) and continuum (Eulerian) models have been developed to understand the hydrodynamics of particle (solid) as well as gas phase. DEM (Discrete Element Modeling) has a wide range of applications in systems involving particles. However, understanding the particle and fluidizing air interactions results in coupling of DEM with a fixed volume description using the Navier Stokes equation for the gas phase.

2. Governing Equation

- **Navier - Stokes equations:** These equations describe the motion of fluid in terms of velocity, pressure and viscosity. They are given as:

$$\partial \rho/\partial t + \nabla \cdot (\rho u) = 0$$

$$\rho (\partial u/\partial t + u \cdot \nabla u) = - \nabla P + \mu \nabla \wedge 2u + F$$

where ρ is the density of fluid, u is the velocity vector, P is the pressure, μ is the dynamic viscosity and F is the external force acting on the fluid.

- **Discrete element method (DEM) equations:** These equations describe the motion of particles in terms of position, velocity and force. They are given as:

$$mi (dvi/dt) = Fi$$

$$ri = ri0 + vi\Delta t + 1/2ai\Delta t \wedge 2$$

where mi is the mass of the particle i, vi is the velocity of the particle i, Fi is the force acting on the particle i, ri is the position vector of the particle i, ri0 is the initial position vector of the particle i, ai is the acceleration of the particle i, and Δt is the time step.

- **Collision models:** These models describe the interaction between two particles in terms of collision frequency, collision diameter, and restitution coefficient. They are given as:

$$fc = \sigma ij|vi - vj| \quad dcoll = di + dj$$

$$e = (vr - v'r) / (u - u')$$

where σij is the collision cross section, vi and vj are the velocities of particles i and j, di and dj are the diameters of particles i and j, e is the restitution coefficient, vr and v'r are the relative velocities of the particles before and after collision, and u and u' are the normal components of relative velocity.

- **Heat transfer equations:** These equations describe the transfer of heat between particles and the fluid. They are given as:

$$qconv = hA (Tp - Tf) \quad qcond = - kA (\nabla Tp)$$

where qconv is the convective heat transfer rate, h is the convective heat transfer coefficient, A is the surface area of the particle, Tp is the temperature of the particle, Tf is the temperature of the fluid, qcond is the conductive heat transfer rate, k is the thermal conductivity of the particle, and ∇ Tp is the gradient of temperature in the particle.

- **Turbulence models:** These models describe the turbulent motion of fluid in terms of Reynolds stresses and turbulence kinetic energy. They are given as:

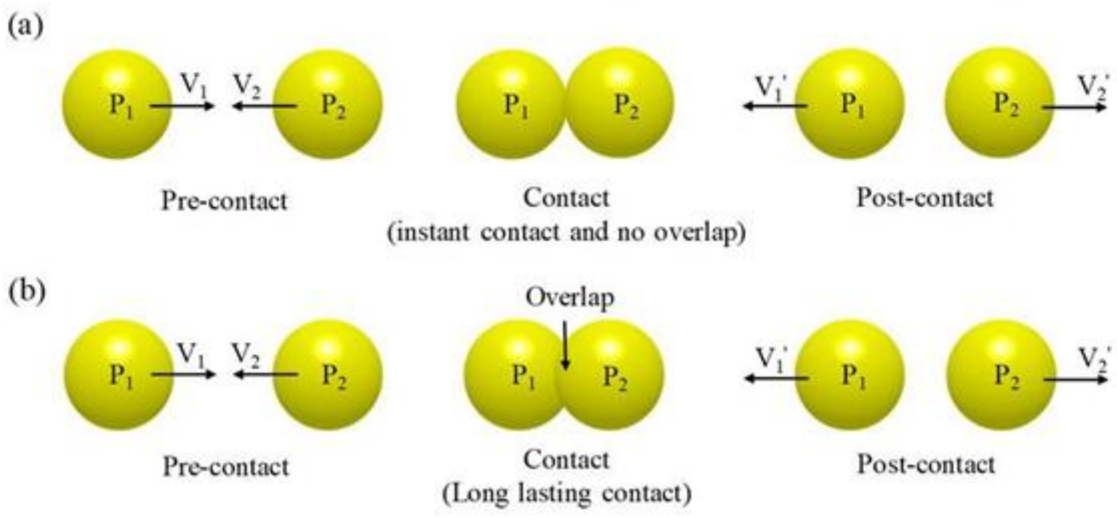
$$\partial (\rho k) /\partial t + \nabla \cdot (\rho ku) = Pk - \varepsilon + \nabla \cdot (\mu eff \nabla k)$$

$$\partial (\rho ui) /\partial t + \nabla \cdot (\rho uiuj) = - \nabla Pi + \nabla \cdot (\mu eff \nabla ui) + \rho g + Ri$$

where k is the turbulence kinetic energy, u is the velocity vector, Pk is the production of turbulence kinetic energy, ε is the dissipation of turbulence kinetic energy, μeff is the effective viscosity, Pi is the pressure, g is the acceleration due to gravity, and Ri is the turbulent Reynolds stress.

3. Collision Frameworks

Hard Sphere Approach: In a hard - sphere system the trajectories of the particles are determined by momentum - conserving binary collisions. The interactions between particles are assumed to be pairwise additive and instantaneous. In the simulation, the collisions are processed one by one according to the order in which the events occur. For not too dense systems, the hard - sphere models are considerably faster than the soft - sphere models. Note that the possible occurrence of multiple collisions at the same instant cannot be accounted



(a) Hard Sphere Approach (b) Soft Sphere Approach

Soft Sphere Approach:

In more complex situations, the behavior of particles may be influenced by short - or long - range forces, and their movements can be determined by integrating Newton's equations of motion. The soft - sphere method, originally developed by Cundall and Strack in 1979, was the first simulation technique for granular dynamics published in the open literature. Soft - sphere models use a fixed time step, allowing particles to slightly overlap. The contact forces are subsequently calculated from the deformation history of the contact using a contact force scheme. The soft - sphere models allow for multiple particle overlap, although the net contact force is obtained from the addition of all pair - wise interactions. These models are essentially time - driven, requiring careful selection of the time step in calculating the contact forces. Various soft - sphere models can be found in literature, differing mainly with respect to the contact force scheme used. Schäfer et al. (1996) presented a review of various popular schemes for repulsive inter - particle forces. Walton and Braun (1986) developed a model using two different spring constants to model the energy dissipation in the normal and tangential directions, respectively. Langston et al. (1994) proposed a force scheme using a continuous potential of an exponential form, containing two unknown parameters: the stiffness of the interaction and an interaction constant.

4. Simulations Results

We did a simulation of rod shaped particle based on MFix - SQP Model and analyzed behavior of fluidized bed at different time interval with initial conditions which is given below

Input and Ouput values of parameters:

Run controls

description = 'fluidized_bed_superdem' run_name = 'FB_SQP'
units = 'SI'
run_type = 'new'
tstop = 2.0
dt = 1.0000e - 02
dt_max = 1.0000e - 02
res_dt = 0.005 #!MFIX - GUI eq{float (1.0/200) }
batch_wallclock = 172800.0
chk_batchq_end =. False.
drag_type = 'SQP_DIFELICE_HOLZER_SOMMERFELD'
turbulence_model = 'NONE' enable_dmp_log =. False.
energy_eq =. False.
nodesi = 1
nodesj = 1
nodesk = 1
term_buffer = 180.0 write_dashboard =. False. full_log =. True. momentum_x_eq (0) =. True. momentum_y_eq (0) =. True. momentum_z_eq (0) =. True. project_version = '9' species_eq (0) =. False. species_eq (1) =. False. species_eq (2) =. False.

Physical parameters

gravity_x = 0.0
gravity_y = - 9.81
gravity_z = 0.0

Cartesian grid cartesian_grid =. False. use_stl =. False.

Numeric

max_nit = 50
norm_g = 0 tol_resid = 1.0000e - 03 ### Discretization
discretize (1) = 0
discretize (2) = 0
discretize (3) = 0
discretize (4) = 0
discretize (5) = 0
discretize (6) = 0
discretize (7) = 0

Geometry

coordinates = 'CARTESIAN'
imax = 10
jmax = 50
kmax = 10
x_max = 0.1
x_min = 0
y_max = 0.5
y_min = 0
z_max = 0.1
z_min = 0

no_k =. False.

Fluid

mu_g0 = 1.8000e - 05 ro_g0 = 1.2005

Solids

mmax = 2
# Solid 1 solids_model (1) = 'SQP'
d_p0 (1) = 0.006442555128342861
ro_s0 (1) = 708.5
nmax_s (1) = 0 e_young (1) = 1.0000e+06 k_s0 (1) = 1.0
ks_model (1) = 'MUSSER' sqp_a (1) = 0.001575
sqp_b (1) = 0.003125
sqp_c (1) = 0.001575
sqp_m (1) = 4.0
sqp_n (1) = 4.0
sqp_q1 (1) = 1.0
sqp_q2 (1) = 0.0
sqp_q3 (1) = 0.0
sqp_q4 (1) = 0.0
v_poisson (1) = 0.3

Initial conditions

Initial condition 1: init1
ic_x_e (1) = 0.1 #!MFIX - GUI eq{float (xmax) } ic_x_w (1) = 0.0 #!MFIX - GUI eq{float (xmin) } ic_y_s (1) = 0.0 #!MFIX - GUI eq{float (ymin) } ic_y_n (1) = 0.1 #!MFIX - GUI eq{float (ymin + 0.1) }
ic_z_b (1) = 0.0 #!MFIX - GUI eq{float (zmin) } ic_z_t (1) = 0.1 #!MFIX - GUI eq{float (zmax) } ic_des_fit_to_region (1) =. False.
ic_ep_g (1) = 0.7
ic_t_g (1) = 293.15
ic_u_g (1) = 0.0
ic_v_g (1) = 0.0
ic_w_g (1) = 0.0
ic_p_star (1) = 0.0
ic_ep_s (1, 1) = 0.3
ic_t_s (1, 1) = 293.15
ic_theta_m (1, 1) = 0.0
ic_u_s (1, 1) = 0.0
ic_v_s (1, 1) = 0.0
ic_w_s (1, 1) = 0.0
ic_ep_s (1, 2) = 0.0
ic_t_s (1, 2) = 293.15
ic_theta_m (1, 2) = 0.0
ic_u_s (1, 2) = 0.0
ic_v_s (1, 2) = 0.0
ic_w_s (1, 2) = 0.0

Boundary conditions

Boundary condition 1: Bottom_inlet bc_type (1) = 'MI'
bc_x_e (1) = 0.1 #!MFIX - GUI eq{float (xmax) } bc_x_w (1) = 0.0 #!MFIX - GUI eq{float (xmin) } bc_y_s (1) = 0.0 #!MFIX - GUI eq{float (ymin) } bc_y_n (1) = 0.0 #!MFIX - GUI eq{float (ymin) } bc_z_b (1) = 0.0 #!MFIX - GUI eq{float (zmin) } bc_z_t (1) = 0.1 #!MFIX - GUI eq{float (zmax) } bc_ep_g (1) = 1.0
bc_p_g (1) = 1.0132e+05 bc_t_g (1) = 293.15
bc_u_g (1) = 0.0
bc_v_g (1) = 5.0
bc_w_g (1) = 0.0
Solid 1
bc_ep_s (1, 1) = 0.0
bc_t_s (1, 1) = 293.15
bc_u_s (1, 1) = 0.0
bc_v_s (1, 1) = 0.0
bc_w_s (1, 1) = 0.0
Solid 2
bc_ep_s (1, 2) = 0.0
bc_t_s (1, 2) = 293.15
bc_u_s (1, 2) = 0.0
bc_v_s (1, 2) = 0.0
bc_w_s (1, 2) = 0.0
Boundary condition 2: Top_outlet bc_type (2) = 'PO'
bc_x_e (2) = 0.1 #!MFIX - GUI eq{float (xmax) } bc_x_w (2) = 0.0 #!MFIX - GUI eq{float (xmin) } bc_y_s (2) = 0.5 #!MFIX - GUI eq{float (ymax) } bc_y_n (2) = 0.5 #!MFIX - GUI eq{float (ymax) } bc_z_b (2) = 0.0 #!MFIX - GUI eq{float (zmin) } bc_z_t (2) = 0.1 #!MFIX - GUI eq{float (zmax) } bc_ep_g (2) = 1.0
bc_p_g (2) = 101325.0
bc_t_g (2) = 293.15
Solid 1
bc_ep_s (2, 1) = 0.0
bc_t_s (2, 1) = 293.15
Solid 2
bc_ep_s (2, 2) = 0.0

VTK outputs

write_vtk_files =. True. time_dependent_filename =. True. vtu_dir = 'VTK'
# VTK output 1: Entire domain vtk_filebase (1) = 'CELL'
vtk_x_e (1) = 0.1 #!MFIX - GUI eq{float (xmax) } vtk_x_w (1) = 0.0 #!MFIX - GUI eq{float (xmin) } vtk_y_s (1) = 0.0 #!MFIX - GUI eq{float (ymin) } vtk_y_n (1) = 0.5 #!MFIX - GUI eq{float (ymax) } vtk_z_b (1) = 0.0 #!MFIX - GUI eq{float (zmin) } vtk_z_t (1) = 0.1 #!MFIX - GUI eq{float (zmax) } vtk_data (1) = 'C'
vtk_dt (1) = 0.01
vtk_nxs (1) = 0
vtk_nys (1) = 0
vtk_nzs (1) = 0 vtk_ep_g (1) =. True. vtk_p_g (1) =. True. vtk_vel_g (1) =. True. # VTK output 3
vtk_filebase (3) = 'SuperDEM_Phase1'

vtk_x_e (3) = 0.1 #!MFIX - GUI eq{float (xmax) } vtk_x_w (3) = 0.0 #!MFIX - GUI eq{float (xmin) } vtk_y_s (3) = 0.0 #!MFIX - GUI eq{float (ymin) } vtk_y_n (3) = 0.5 #!MFIX - GUI eq{float (ymax) } vtk_z_b (3) = 0.0 #!MFIX - GUI eq{float (zmin) } vtk_z_t (3) = 0.1 #!MFIX - GUI eq{float (zmax) } vtk_data (3) = 'P'
vtk_dt (3) = 0.01
vtk_nxs (3) = 0
vtk_nys (3) = 0
vtk_nzs (3) = 0 vtk_part_diameter (3) =. True. vtk_part_vel (3) =. True. vtk_part_phase (3, 1) =. True. vtk_part_phase (3, 2) =. False. # VTK output 4
vtk_filebase (4) = 'SuperDEM_Phase2'
vtk_x_e (4) = 0.1 #!MFIX - GUI eq{float (xmax) } vtk_x_w (4) = 0.0 #!MFIX - GUI eq{float (xmin) } vtk_y_s (4) = 0.0 #!MFIX - GUI eq{float (ymin) } vtk_y_n (4) = 0.5 #!MFIX - GUI eq{float (ymax) } vtk_z_b (4) = 0.0 #!MFIX - GUI eq{float (zmin) } vtk_z_t (4) = 0.1 #!MFIX - GUI eq{float

(zmax) } vtk_data (4) = 'P'
vtk_dt (4) = 0.01
vtk_nxs (4) = 0
vtk_nys (4) = 0
vtk_nzs (4) = 0 vtk_part_diameter (4) =. True. vtk_part_vel (4) =. True. vtk_part_phase (4, 1) =. False. vtk_part_phase (4, 2) =. True.
SPx outputs spx_dt (1) = 100.0
spx_dt (2) = 100.0
spx_dt (3) = 100.0
spx_dt (4) = 100.0
spx_dt (5) = 100.0
spx_dt (6) = 100.0
spx_dt (7) = 100.0
spx_dt (8) = 100.0
spx_dt (9) = 100.0
spx_dt (10) = 100.0
spx_dt (11) = 100.0

Residuals group_resid =. True. resid_string (1) = 'P1' resid_string (2) = 'U1' resid_string (3) = 'V1'

Discrete element model des_coll_model = 'HERTZIAN' des_en_input (1) = 0.5
des_en_input (2) = 0.5
des_en_input (3) = 0.5
des_en_wall_input (1) = 0.5
des_en_wall_input (2) = 0.5
des_epg_clip = 0.1
des_et_input (1) = 0.5
des_et_input (2) = 0.5
des_et_input (3) = 0.5
des_et_wall_input (1) = 0.5
des_et_wall_input (2) = 0.5
des_etat_fac = 0.5
des_etat_w_fac = 0.5 des_interp_mean_fields =. True. des_interp_on =. True.
des_interp_scheme = 'DPVM_SATELLITE'
des_neighbor_search = 4
desgridsearch_imax = 10
desgridsearch_jmax = 50
desgridsearch_kmax = 10 ew_young = 1.0000e+06 gener_part_config =. True.
kt_fac = 0.28571428571429 #!MFIX - GUI eq{float (2/7) }
kt_w_fac = 0.28571428571429 #!MFIX - GUI eq{float (2/7) } mew = 0.4
mew_w = 0.3
neighbor_search_n = 20
neighbor_search_rad_ratio = 1.0
nfactor = 0
vw_poisson = 0.3

Two - fluid model
friction_model = 'SCHAEFFER' Output Results:
Results for the above condition solids of superquadric nature which is rod shaped particle at different time interval is captured and shown below, from violet to yellow is the color map for the velocity ranges of the particle and it can be clearly observed that particles at higher positions of reactor and have higher velocities. Here we have used MFix - SQP Model and solving for superquadric particle, with SQP DiFelice - Holzer - Sommerfeld as the drag model with a mesh of 10, 50, 10 respectively in x, y and z direction. In initial condition we have considered 0.3 as the volume fraction of solid particles and 0.7 as the volume fraction of fluid from bottom to 0.1m in y - axid in a geometry of 0.1*0.5*0.5 m. and the output shown considers volume fraction, pressure and velocity vector of Fluid and the results are as shown below