Coupled CFD-DEM Model for a Pneumatic Conveying System

Jacob R. Madden
jrmadden@usc.edu

Viterbi School of Engineering
University of Southern California
ASTE 499 - Dr. Lubos Brieda

May 13, 2020

Introduction

- Review of the CFD-DEM modeling approach for a continuum fluid- discrete particle system
- Summary of fluid and particle phase governing equations used for CFD-DEM simulations
- Detailed description of modeling approach, discretization techniques, and model parameters
- Presentation of preliminary simulation results and validation with comparable analyses
- Summary of completed work and applications, discussion of future numerical modeling work

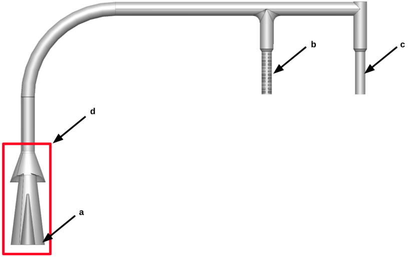

Figure 1: Physical system used for simulation of numerical model: (a) Inlet annulus (b) Outlet venting to vacuum (c) Secondary chamber (no vents) (d) Reduced system geometry section

CFD-DEM Modeling

- Numerical modeling of two-phase systems can be generally categorized by the length scales of the fluid and particle states considered
- Formulation of conservation equations and closure laws depend on whether the states are taken as discrete or continuum, and the resultant averaging assumptions.
- Three main approaches (for varying scales): two-fluid model (TFM), direct simulation-discrete element (DNS-DEM), and computational fluid dynamics-discrete element (CFD-DEM) [1]
- **TFM**: Assumes both fluid and particle are in the meso-scale and of continuum phase (Eularian-Eularian); reduced computational workload, but less accurate model of bulk particle motion due to semi-empirical relations
- **DNS-DEM**: surface stresses on each particle due to fluid flow are directly computed using Navier-Stokes, meaning that empirical correlations are not required to capture drag and lift; extremely accurate, but also computationally expensive
- **CFD-DEM**: continuum fluid mesh with discrete particle dynamics (Eularian-Lagrangian); compute fluid properties across each cell and directly resolve particle dynamics, with momentum exchange between phases; balances accurate resolution of particle dynamics with computational workload

Governing Equations (1/2)

- There are two primary models accepted for use in CFD-DEM modeling are referred to as Model A or Model B [1]
- General governing equations for the particle phase can be seen in Eqn. (1) and (2). These describe the translational and rotational motion of each particle, where particle-fluid interaction force is given by f_{pf}, inter-particle elastic and damping forces are given by f_c and f_d, respectively, and inter-particle tangential moments and and rolling friction are given by M_t and M_r.
- Note that the particle-fluid interaction force, f_{pf}, has a different formulation depending on the model used.

$$m_i \frac{d\mathbf{v}_i}{dt} = \mathbf{f}_{pf,i} + \sum_{j=1}^{k_c} (\mathbf{f}_{c,ij} + \mathbf{f}_{d,ij}) + m_i \mathbf{g} \tag{1}$$

$$I_i \frac{d\omega_i}{dt} = \sum_{j=1}^{k_c} (\mathbf{M}_{t,ij} + \mathbf{M}_{r,ij}) \tag{2}$$

Governing Equations (2/2)

- The Navier-Stokes equations govern flow in the fluid phase, and under the continuum phase assumption here, governing equations are derived using a local averaging method over each cell, resulting in mass conservation and momentum conservation equations (Eqn. 4 and 3).
- Then, the formulation for the particle-fluid interaction forces as described in Model A are given by Eqn. (6) and (5), where $\mathbf{f}_d$ is drag force, $\mathbf{f}_{\nabla p}$ is pressure gradient force, $\mathbf{f}_{\nabla \tau}$ is viscous force force, and f'' is the sum of non-dominant particle-fluid interaction forces, such as Basset force, Saffman force, Magnus force, etc [1].

$$\delta(\epsilon_f)/\delta t + \nabla \cdot (\epsilon_f \mathbf{u}) = 0 \tag{3}$$

$$\delta(\rho_f \epsilon_f \mathbf{u})/\delta t + \nabla \cdot (\rho_f \epsilon_f \mathbf{u}\ \mathbf{u}) = -\epsilon_f \nabla p - \mathbf{F}_{pf} + \epsilon_f \nabla \tau + \rho_f \epsilon_f \mathbf{g} \tag{4}$$

$$\mathbf{F}_{pf} = \frac{1}{\Delta V}\left(\mathbf{f}_d + \mathbf{f''}\right) \tag{5}$$

$$\mathbf{f}_{pf} = \mathbf{f}_d + \mathbf{f}_{\nabla p} + \mathbf{f}_{\nabla \tau} + \mathbf{f''} \tag{6}$$

Modeling Implementation (Software)

- **OpenFOAM** (Open-source Field Operation and Manipulation): C++ based library used to model the CFD sub-model - base solvers and algorithms that an be heavily modified (discretization schemes, turbulence, thermophysical properties, solver algorithm, meshing applications, etc)
- **LIGGGHTS** (LAMMPS improved for general granular and granular heat transfer simulations): open-source, classical molecular dynamics simulator for DEM - definition of particle mechanical properties, cohesion and contact models, external force fields, etc
- **CFDEMcoupling**: open-source framework utilizing MPI to decompose CFD and DEM domains, and transfer momentum data between solvers (two-way coupling)

Simulation Model (CFD 1/2)

- PISO algorithm (Pressure Implicit with Splitting of Operators) used to iteratively solve Navier-Stokes
- Fluid pressure and velocity are the dependent variables, and the “splitting” refers to breaking the solutions process into steps which decouple the operations in the discretized momentum and pressure equations [2]
- First, the discretized momentum equation is solved implicitly to compute an intermediate velocity field using the current pressure field (the “momentum predictor” step)
- Then, a pressure correction equation is solved using this velocity field, and then used to correct the velocities (“corrector” step one)
- Then, the updated pressure and velocity fields are used along with the intermediary fields in a second correction loop to solve for the final pressure and velocities (“corrector” step two)
- Under-relaxation of the velocity and pressure correction factors promote convergence, so a balance must be struck between computational time and accuracy.

Simulation Model (CFD 2/2)

- Time discretization uses Crank-Nicolson scheme: second-order, implicit finite-difference method which is unconditionally stable
- All other terms (except for k-ϵ) are discretized using a standard finite-volume scheme with linear interpolation of values from cell centers to faces to calculate flux
- To improve solver stability, the k-ϵ terms in the turbulence model were discretized using a bounded upwind method (reduction of accuracy but ensures stability)
- CFD timestep calculated using Courant-Friedrichs-Lewy (CFL) condition: non-dimensional Courant number must be less than unity to resolve fluid flow characteristics

$$C = \frac{u\Delta t}{\Delta x} \tag{7}$$

Simulation Model (DEM 1/2)

- Granular interaction model described by Eqn.(8): defines particle-particle forces between neighboring particles [3]
- Characterized by a normal force (consisting of contact and damping forces) and a tangential force (consisting of shear and damping forces) between particles
- Defined by elastic and viscoelastic material constants to define the contact and damping forces
- Cohesion effect is accounted for using a simplified Johnson-Kendall-Roberts model in Eqn. (9), which adds an additional term to the normal force interaction [4]

$$F = (k_n \delta n_{ij} - \gamma_n v n_{ij}) + (k_t \delta t_{ij} - \gamma_t v t_{ij}) \tag{8}$$

$$F_n = k\,(2\pi\delta_n(2R)) \tag{9}$$

Simulation Model (DEM 2/2)

- DEM time step is defined by two time step criteria, which are derived from the particle collision event, and both of which are defined for the Hertzian contact model used here [5]
- Rayleigh criteria based on the time required for elastic wave propagation occuring during particle collisions, per Eqn. (10)
- Second is the Hertz time, defined by the theoretical duration of contact between two spherical particles predicted by the Hertz-Mindlin contact theory, per Eqn. (11)

$$dt_r = \pi r \frac{\sqrt{\frac{\rho}{g}}}{0.1631\nu + 0.8766} \tag{10}$$

$$dt_h = 2.87 \left(\frac{m_{eff}^2}{r_{eff} Y_{eff}^2 v_{max}} \right)^2 \tag{11}$$

Simulation Model (Coupling)

- Two-way coupling between fluid and particle phases occurs by way of momentum exchange
- As previously discussed, the particle-fluid interactions forces (drag, pressure gradient, and viscous) appear in both CFD and DEM momentum conservation equations
- Fluid phase conservation equation also contains particle void fraction and particle-velocity terms by way of an implicit momentum source term [6]

$$K_{pf} = \frac{\alpha_f \mid \sum_i \mathbf{F_{pf}} \mid}{V_{cell} \mid \mathbf{u_f} - \mathbf{u_p} \mid} \tag{12}$$

- Since the initial stages of the model is approximated as fluidized bed, the Koch-Hill drag model was implemented, per Eqn. (13) [7]

$$\beta = \frac{18\mu_f \epsilon_f^2 \epsilon_p}{d_p^2} \left(F_0(\epsilon_p) + 0.5 F_3 \epsilon_p Re_p \right) \tag{13}$$

Simulation Results

- Two testing geometries used: reduced portion of the full system and the full system, per Fig. (1)
- Reduced system used to verify (1) base solver stability, (2) quantitative accuracy of the outlet fluid properties as compared with a separate compressible flow analyses (not described here), and (3) qualitative accuracy of the bulk particle movement as compared with experiments
- Full system simulated to verify (1) the quantitative accuracy of the particle velocity profile in the downstream tube section as compared to experiments, and (2) downstream fluid velocity trends as compared to compressible flow analyses
- Convergence study completed on reduced section to verify mesh- and timestep-independence of base algorithm and simulation
- Sensitivity analysis completed to analyze how the model outputs changed as a function of the particle coefficient of restitution, alphaCOR (not easily characterized)
- To validate the incompressible simulation results, analytical calculations using compressible fanno flow analysis and a compressible flow simulation (using the commercial CFD solver, Ansys Fluids) were completed
- These analyses indicated that fluid density across the reduced system geometry remained approximately constant, so the incompressible simulation should be valid

Simulation Results (Reduced Geometry 1/2)

- Overall: high stability and convergence towards the expected velocity profile when using the fluid density predicted from the compressible flow analyses
- Mean steady-state outlet velocity is approximately 60-70 m/s, which agrees with that predicted by the compressible flow analyses (55-65 $\frac{m}{s}$ m/s), validating the base CFD sub-model. (Both reduced and full system geometries are showed in Fig 2 for comparison).

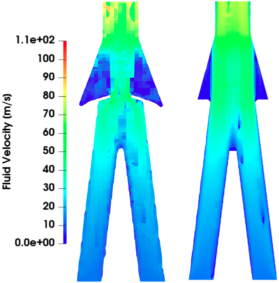

Figure 2: Annulus velocity profile at 0.125 seconds for full system (left) and reduced geometry (right)

Simulation Results (Reduced Geometry 2/2)

- The predicted bulk particle movement also closely matched the experimentally observed transport behavior.
- The bed immediately fluidizes and all of the particles are evacuated out of the annulus chamber, with some particles cohesively attaching to the outer walls of the chamber and the intermediary outlet ledge

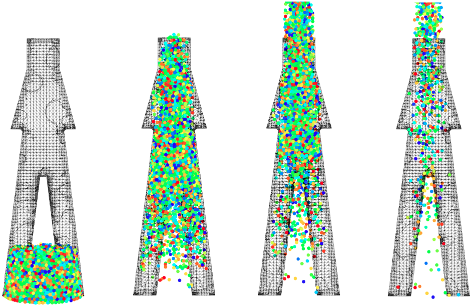

Figure 3: Particle behavior at t = 0.001, 0.05, 0.08, 0.125 seconds

Simulation Results (Full System)

- Axial particle velocity through the downstream tube reached 5-6 m/s, also matching with the experimental data velocity data (calculated using image processing of testing videos)
- Downstream fluid velocity trends also agreed with the compressible flow simulations, showing a sharp acceleration immediately before the outlet past the branch (though not matching exactly due to the incompressible flow assumption)
- Finally, the bulk particle movement and distribution into the downstream chambers agreed with the experimental data.

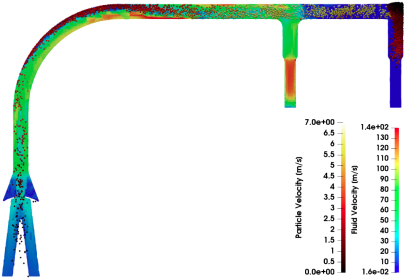

Figure 4: Downstream particle distribution and velocity profile

Simulation Results (Convergence Study 1/2)

- To verify that this model is mesh- and timestep- independent, a convergence study was completed to analyze the transient pressure gradient and velocity trends in the reduced geometry system
- To verify mesh-independence, "coarse" and "fine" CFD domains were generated by varying resolution of the base hex block domain. The coarse mesh used approximately 5e4 cells, while the fine mesh used around 5e5 cells. Both ran with the same timestep (2e-5 sec.)
- The transient velocity profiles of both mesh types can be seen in Fig 5. Both simulations showed similar qualitative behavior, but the finer mesh exhibited a "delay", likely due to increased time required to fully capture fluid velocity in the particle-fluid momentum balance
- Because the main model output of interest was particle behavior, this was deemed to be acceptable.

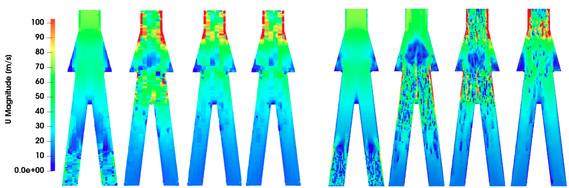

Figure 5: Velocity profile [$\frac{m}{s}$] at t = 0.001, 0. 05, 0.08, 0.125 sec for coarse (L) and fine (R) mesh

Simulation Results (Convergence Study 2/2)

- In testing timestep-independence, the large and small timesteps were set to $dt = 1e - 4$ and $dt = 5e - 6$ seconds, respectively. In the small timestep case, the mean Courant number hovered around 0.18, while the mean Courant number for the large timestep remained stable around 0.96.
- The transient pressure gradient of both mesh types can be seen in Fig 6. Both models showed very similar qualitative results, with only minor variations (<5-10%) as the bulk particle mass moved out of the initial annulus, indicating that the model is relatively timestep-independent.

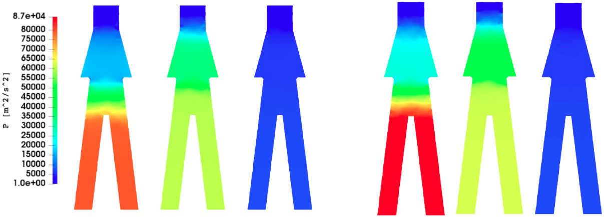

Figure 6: Fluid pressure [$\frac{m^2}{s^s}$] at t=0.05, 0.08, 0.125 sec. for $dt = 1e - 4$ (L) and $dt = 1e - 6$ (R) sec.

Simulation Results (Sensitivity Analysis)

- To test the model dependence on particle properties, the particle coefficient of restitution, α_{COR}, was varied between 0.1 and 0.5. This property varies based on material type, impact velocity, and impact history, so it is difficult to characterize.
- Steady-state particle distribution was recorded for the full-system model (since that is the primary simulation output of interest in regards to particle data)
- Tests showed extremely low deviation between the high and low α_{COR} values. Representative collections can be seen in Fig 7.

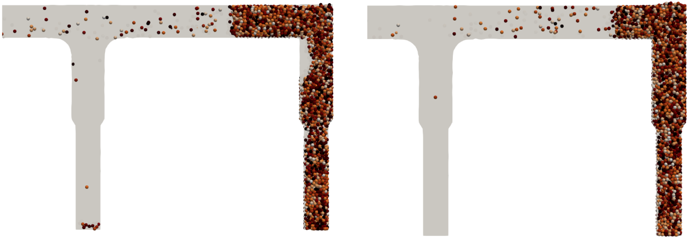

Figure 7: Particle distribution at t=0.125 sec. for $\alpha_{COR} = 0.1$ (L) and $\alpha_{COR} = 0.5$ (R)

Conclusion and Future Work

- Preliminary results indicate that this low-fidelity model successfully validates hardware testing, providing a numerical tool which can be used to significantly accelerate the pneumatic system design workflow by reducing the physical testing required to iterate on designs. However, much work remains to be completed.
- Challenges:
  - Deriving governing equations and their discretized forms for continuum fluids (not trivial!)
  - Recognizing capabilities and limitations of different discretization schemes (ie. upwind, linear upwind, LUST, vanLeer, cubic, etc)
  - Verifying and testing solver stability conditions
- Potential future work:
  - Convert full-system CFD to a compressible flow model, to better capture downstream physics
  - Implement updated drag force model for particle-fluid interaction force in downstream tube section (Koch-Hill primarily used in fluidized beds)
  - Optimize JKR particle cohesion model parameters for the specific particle type used here

ZY Zhou, SB Kuang, KW Chu, and AB Yu.
Discrete particle simulation of particle–fluid flow: model formulations and their applicability.
Journal of Fluid Mechanics, 661:482–510, 2010.

Raad I Issa.
Solution of the implicitly discretised fluid flow equations by operator-splitting.
Journal of computational physics, 62(1):40–65, 1986.

pair_style gran command.
`https://www.cfdem.com/media/DEM/docu/pair_gran.html`.
[Online; accessed 12-May-2020].

Gran cohesion sjkr2 model.
`https://www.cfdem.com/media/DEM/docu/gran_cohesion_sjkr2.html`.
[Online; accessed 12-May-2020].

Shane J Burns, Petri T Piiroinen, and Kevin J Hanley.
Critical time step for dem simulations of dynamic systems using a hertzian contact model.
International Journal for Numerical Methods in Engineering, 119(5):432–451, 2019.

K. Kloss.
Models, algorithms and validation for opensource dem and cfd-dem.
Progress in Computational Fluid Dynamics, 12(2/3):140 – 252, 2012.

Maureen S Van Buijtenen, Willem-Jan Van Dijk, Niels G Deen, JAM Kuipers, T Leadbeater, and DJ Parker.
Numerical and experimental study on multiple-spout fluidized beds.
Chemical engineering science, 66(11):2368–2376, 2011.

Piso algorithm.
`https://openfoamwiki.net/index.php/PISO`.
[Online; accessed 12-May-2020].

Backup

- Example of PISO algorithm (Pressure Implicit with Splitting of Operators) for one-dimensional flow [8]
- Simplified momentum equation:

$$\frac{\delta u}{\delta t} + \frac{\delta}{\delta x}(uu) = -\frac{\delta p}{\delta x} + \rho g \tag{14}$$

- Step 1: Predictor step (using backward Euler to simplify notation), with interpolation from cell centers to faces, where n denotes the current time step and * denotes the "predicted" values, and dividing through by cell volume:

$$\left[\frac{1}{\Delta t}\left(\frac{u^n_{i+1/2} - u^n_{i-1/2}}{2\Delta x}\right)\right] u^*_i + \left(\frac{u^n_{i+1/2}}{2\Delta x}\right) u^*_{i+1} - \left(\frac{u^n_{i-1/2}}{2\Delta x}\right) u^*_{i-1} = \frac{u^n_i}{\Delta t} - \left(\frac{\delta p}{\delta x}\right)|^n_i + (\rho g)|^n_i \tag{15}$$

- Using matrix-vector notation, and expanding the coefficient matrix on the left hand side to a diagonal matrix, A, and off diagonal matrix, H', this simplifies to:

$$Au^* + H'u^* = r - \nabla p^n + \rho g^n \tag{16}$$

Backup

- Step 2: Corrector step, first generating an equation for the corrected velocity, u^{**} in terms of predicted velocity, u^*, old velocity, u^n, and corrected pressure, p^*.

$$\left[\frac{1}{\Delta t}\left(\frac{u^n_{i+1/2} - u^n_{i-1/2}}{2\Delta x}\right)\right] u_i^{**} + \left(\frac{u^n_{i+1/2}}{2\Delta x}\right) u^*_{i+1} - \left(\frac{u^n_{i-1/2}}{2\Delta x}\right) u^*_{i-1} = \frac{u^n_i}{\Delta t} - \left(\frac{\delta p}{\delta x}\right)|^*_i + (\rho g)|^n_i \tag{17}$$

- Again using matrix-vector notation, this simplifies to:

$$Au^{**} + H' u^* = r - \nabla p^* + \rho g^n \tag{18}$$

- Taking the divergence of this equation and recognizing that $\nabla u^{**} = 0$ by continuity results in the below equation, which can be solved for p^*. Then, this corrected pressure can be used to solve for corrected velocity in Eqn. (18).

$$\nabla^2 \left(A^{-1} p^*\right) = \nabla \cdot \left(A^{-1} H + A^{-1} \rho g^n\right) \tag{19}$$

- Step 3: Further corrector steps can be applied using the same matrix formulation with A and H (defined as $H = r - H' u^*$. An additional corrector steps would use the following equations, where p^{**} and u^{***} are the second corrected pressure and velocity:

$$\nabla^2 \left(A^{-1} p^{**}\right) = \nabla \cdot \left(A_{-1} H + A^{-1} \rho g^n\right) \tag{20}$$

$$u^{***} = A^{-1} H + A^{-1} \nabla p^{**} + A^{-1} \rho g^n \tag{21}$$