

NSF CDSE DRAFT 02

Thor, Gibbs, Zeus, the guy who invented sticky notes, Santa Claus*

*North Pole department of ice and snow Fwding address - Department of Chemical and Biochemical Engineering,
Rutgers, The State University of New Jersey, Piscataway, NJ, USA 08854*

Abstract

abstract text goes here ...

Keywords: Population balance model, heteroaggregation, alginate, chitosan, oppositely charged

1. Introduction & Objectives

description of particulate processes and challenges... leads up to our "solution" or start of solution which is what this paper is about
particulate process and how they are half all chem industry
despite being important they are VERY hard to model and study bec of microphenomena.
strain difficulty and added costs to industry bec of this
PBM DEM have been helpful for this however but they take long time to solve
in the past when faced with comp heavy tasks scientists have used parallel computing. divide big prob into small pieces that are simultaneously solved. time to solve is time to solve small piece.
potential benefits this could have 1. unprecedeted accuracy 2.wide adoption of using these models since they are so fast 3. cheaper drugs 4. MPC accurate but in real time 5. parameter estimation for QbD 6. etc.

1.1. Objectives

1. develop 4D PBM that is parallelized spatially and internally for optimal utilization of modern HPC equipment.
2. use liggghts to accurately model micromechanics for processing in Lodgitech granulator.
3.LINK PBM-DEM for best of both worlds and optimal performance. 1-way or if possible 2-way coupling

19 2. Background & Motivation

20 2.1. Particulate Processes

pharma / particulate processes information prevalence etc
difficult physics bec micro phenomena etc
as result use time consuming and expensive heuristics or inefficient operation protocol dotdotdot
also reason most blah done in batch mode still bec continuous mode compounds complexity of these systems

*Corresponding author

Email address: **santaclaus** (Santa Claus)

26 will need to switch to continuous etc
 27 segway with need to better understand these systems from a modelling/theory point of view
 28 dotdotdot QbD etc

29 *2.2. Modeling*

30 *2.2.1. Discrete Element Modeling (DEM)*

Discrete Element Method is a simulation technique used to monitor the behaviour of each particle as a separate entity compared to other bulk continuum models. This method tracks the movement of each of the particles with in the space, records the collisions of each particle with the geometry as well as with each other and it is also subject to other force fields like gravity (Barrasso and Ramachandran (2015)) This model is based on the Newton's laws of the motion and is expressed as in Equations 1 and 2 :

$$m_i \frac{dv_i}{dt} = F_{i,net} \quad (1)$$

$$F_{i,net} = F_{i,coll} + F_{i,ext} \quad (2)$$

31 In the above equations m_i represents the mass of the particle, v_i represents the velocity of the
 32 particle, $F_{i,net}$ represents the net force on the particle, forces on the particle due to collisions and
 33 other external forces are represented in $F_{i,coll}$ $F_{i,ext}$ respectively.

34 The distance between each particle calculated at every time step and if the distance between
 35 two particles is less than the sum of the radii (for spherical particles) a collision between the two
 36 particles is recorded. The tolerance for overlap is low in the normal as well as the tangential direction
 37 (Cundall and Strack (1979)). Microscale DEM simulations are computationally demanding and
 38 simulations may take upto several days to replicate a few seconds of real time experiments. Many
 39 methods have been implemented to increase the speed of these simulations, such as scaling by
 40 increasing the size of the particles. These approximations are good in understanding the physics
 41 of the system but are not directly applicable to process-level simulations.

42 Thus, this method for granular powder is usually replaced by Population Balance Method
 43 (PBM) which is a much quicker approximation as it is a bulk model.

44 *2.2.2. PBM*

45 Population balance models predict how groups of discrete entities will behave on a bulk scale due
 46 to certain effects acting on the population with respect to time (Ramkrishna and Singh (2014)). In
 47 the context of process engineering and granulation, population balance models are used to describe
 48 how the number densities, of different types of particles, in the granulator change as rate processes
 49 such as aggregation and breakage reshape particles (Barrasso et al. (2013)). In a general form of
 50 population balance model is shown here as equation 3.

$$\begin{aligned} \frac{\partial}{\partial t} F(\mathbf{v}, \mathbf{x}, t) + \frac{\partial}{\partial \mathbf{v}} [F(\mathbf{v}, \mathbf{x}, t) \frac{d\mathbf{v}}{dt}(\mathbf{v}, \mathbf{x}, t)] + \frac{\partial}{\partial \mathbf{x}} [F(\mathbf{v}, \mathbf{x}, t) \frac{d\mathbf{x}}{dt}(\mathbf{v}, \mathbf{x}, t)] \\ = \mathfrak{R}_{formation}(\mathbf{v}, \mathbf{x}, t) + \mathfrak{R}_{depletion}(\mathbf{v}, \mathbf{x}, t) + \dot{F}_{in}(\mathbf{v}, \mathbf{x}, t) - \dot{F}_{out}(\mathbf{v}, \mathbf{x}, t) \end{aligned} \quad (3)$$

51 In equation 3 \mathbf{v} is a vector of internal coordinates. In a granulation system \mathbf{v} is commonly
 52 used to describe the solid, liquid, and gas content of each type of particle. The vector \mathbf{x} represents
 53 external coordinates which account for spatial variance in the population density in the granulation
 54 system.

55 talk about kernels - empirical, semi mech, and DEM informed? talking about kernels then
56 mentioning there is DEM informed could be GREAT SEGWAY to multi-physics models dotdotdot
57 combined PBM DEM etc.

58 *2.2.3. multi-physics models*

59 general goal of mutli physics is best of two (or more worlds) accuracy of one but the time savings
60 of another etc.

61 finish with PBM DEM goals and past works? then mention it still can take a long time dotdotdot
62 SEGWAY into parallel computing?

63 *2.3. Parallel Computing and Computer Architectures*

64 *2.3.1. Overview*

65 The goal of parallel computing is to distribute large amounts of computation across many
66 compute cores to solve problems faster (Wilkinson and Allen (2005)).

67 *2.3.2. Computer Architecture*

68 Parallel programs are commonly run on computer clusters. Computer clusters are a collection
69 of nodes interconnected by a high speed communication network for message passing from one node
70 to another. Each node has RAM and one or more CPUs, each CPU often has multiple compute
71 cores. Commonly nodes are manufactured with two CPUs, each CPU usually has around 8-16
72 cores **maybe too big of an assumption**. CPUs have their own memory called cache which is much
73 faster than RAM which is why cache utilization should be favoured over RAM if possible **phrasing?**
74 RAM is often split between the CPU sockets, so each CPU has a direct connection to a RAM bus.
75 This means that if there are two CPUs on a node they will not be able to access the RAM that is
76 connected to the other CPU as quickly as the RAM it is directly connected to Jin et al. (2011).

77 Computer architectures are often classified by these memory locality dynamics. There are
78 two distinct classes, distributed memory systems and shared memory systems. A cluster is a
79 combination of the two classes. Each node operates its memory independently of the other nodes
80 and explicit message passing is needed to share memory between nodes. While the cores on a node
81 operate in shared memory with each other since message passing is not explicitly needed to update
82 the memory each core as access too. All of these aspects of the computer architecture should be
83 considered when designing a parallel program for the best performance Adhianto and Chapman
84 (2007) **maybe reword better to make more like CACE paper. should I state network speed much**
85 **slower than RAM so try to send few messages as possible etc? could be more explicit in these**
86 **statements?**

87 *2.3.3. Parallel Application Program Interfaces*

88 Message Passing Interface (MPI) is a common parallel computing application interface standard.
89 MPI is used for distributed memory parallel computing, this is because MPI will operate every
90 MPI process as a discrete unit that does not share memory with the other processes unless explicit
91 message passing is used. Even on shared a single node where the hardware is supports shared
92 memory computing, MPI will still operate it in a distributed memory fashion Jin et al. (2011).
93 Operating all cores as distinct units also means they each need their own copy of all variables used
94 for computation which results in a large overall memory foot print compared to a same system if
95 it was operated in shared memory.

96 Open Multi-Processing (OMP) is another application program interface stand for parallel com-
97 puting. OMP is used for shared memory and can take advantage of shared memory systems which
98 can result in much faster computation. It does not work well on distributed systems though. This

99 prevents it from being used to efficiently carry out computations across multiple nodes of a cluster
100 simultaneously Jin et al. (2011).

101 Since MPI is best for distributed computing and OMP is better for shared computing many
102 individuals have studied the performance of MPI vs MPI+OMP methods and many studies have
103 used MPI+OMP for scientific computation for improved performance. Often times a trade off is
104 made between optimizing a program for performance and trying to make it flexible enough to run
105 on many different computer architectures [might need reference for this](#). A summary of some works
106 addressing MPI+OMP methods for scientific computing and architecture features and concerns can
107 be found in Bettencourt et al. (2017). In the conclusion to the work by Bettencourt et al. (2017)
108 it was found that hybrid methods for PBMs allow the code flexibility for different architectures
109 while still maintaining good performance. [should I mention load balancing techniques of gunawan](#)
110 [paper?](#). In the work of Bettencourt et al. (2017) only the external coordinates of the PBM were
111 parallelized. In this current work external and internal calculations are parallelized.

112 3. Methods

113 3.1. DEM

114 3.1.1. LIGGGHTS

115 LIGGGHTSv3.60 (Kloss et al. (2012)) developed by DCS computing was used for all the simu-
116 lation performed in this study. Edits were made to the compute_contact_atom source file to obtain
117 particle – particle collisions. The aforementioned version of LIGGGHTS was compiled using the
118 mvapich (mvapich2 v2.1) and intel (intel v15.0.2) compilers with the -o3 optimisation option as
119 well as an option to use OpenMP threads was implemented. This hybrid parallel technique helped
120 achieve significant speed improvements over MPI only compilations. The speed improvements are
121 illustrated in Table (refer to the speed table). The studies were performed on STAMPEDE super-
122 computer located at University of Texas, Austin. The hardware configuration of each node consists
123 of 2 8-core Intel Xeon E5-2680 processors based on the Sandy Bridge architecture, 32 gb of memory
124 with QPI interconnects at 8.0 GT/s PCI-e lanes.

125 3.1.2. Geometry

126 [check other file Charles uploaded to see if that one was more "journal ready"](#)

127 In this study, the Lödige CoriMix CM5 continuous high shear granulator has been studied. Its
128 geometry was developed using the SolidWorksTM (Dassault Systèmes). This granulator consisted
129 of a high speed rotating element enclosed within a horizontal cylindrical casing. The casing (shown
130 in Figure 1) consists of a cylinder with diameter of 120 mm at the inlet and 130 mm at the outlet
131 and having a total length of 440 mm. A vertical inlet port is provided at one end of the casing and
132 an angled outlet port is provided at the larger end of the case.

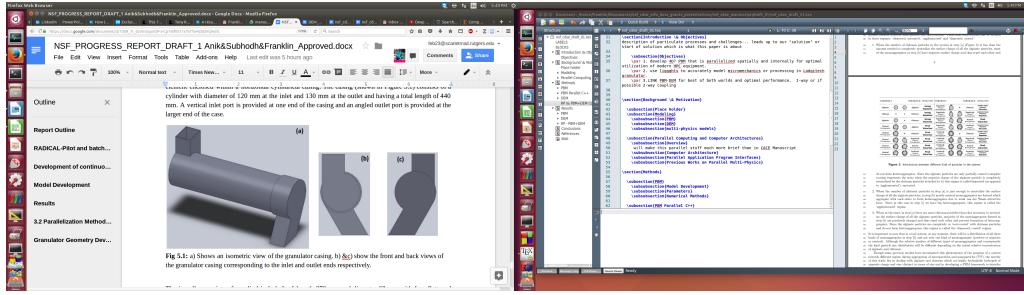


Figure 1: a) Shows an isometric view of the granulator casing. b) and c) show the front and back views of the granulator casing corresponding to the inlet and outlet ends respectively.

133 The impeller consists of a cylindrical shaft of length 370 mm and diameter 68 mm with four
 134 flattened sides 15 mm wide running along the axis. The blades are placed on these flattened sides
 135 as shown in figure 2. There are three different blade elements on the shaft (figure 2). At the
 136 granulator inlet, there are 4 paddle shaped feed elements following which there are 20 tear drop
 137 shaped shearing elements and finally 4 trapezoidal blades near the exit. All these elements are
 138 placed in a spiral configuration. The final configuration of the granulator is shown in figure 3.

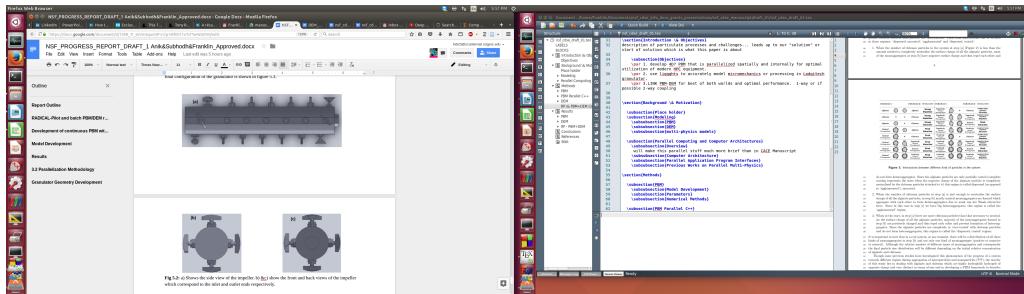


Figure 2: a) Shows the side view of the impeller. b) and c) show the front and back views of the impeller which correspond to the inlet and outlet ends respectively.

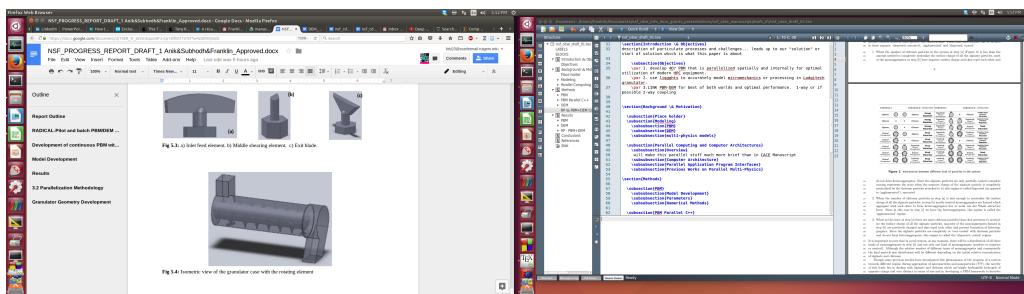


Figure 3: a) Inlet feed element. b) Middle shearing element. c) Exit blade. blah blah Isometric view of the granulator case with the rotating element

139 3.1.3. Meshing

140 After the geometry was built in solidworks the shell and impeller were exported as **non-binary**
 141 STL files. The coarsest output options(**include settings even if they were automatic may change**
 142 **from one version to another of SW**) were used to keep the STL files small and simple for faster
 143 computations times (**needs a citation? or is common knowledge?**). They were also exported not

144 keeping there original coordinates (1.too much info 2. what was option in SW for that so I can
145 use that wording) This resulted in the impeller having x-number-faces and y-number-points with
146 approximately a file size of number KBs. The shell had x-number-faces and y-number-points and
147 approx number KBs.

148 Meshlab was used to align the STL files for importing into LIGGGHTS. No mesh treatments
149 were used on the STLs.

150 The meshes were then imported into LIGGGHTS using the write command (more specific com-
151 mand that we actually used?). This resulted in 50 elements of the impeller file having "highly
152 skewed elements angles $| 0.5$ deg?" that according to LIGGGHTS would degrade parallel perfor-
153 mance. The shell did not have any skewed elements (FUTURE SOLUTION? - perhaps we can use
154 the output from liggghts exclusion list to find exact elements of issue. then we can use meshlab to
155 exclude those peices or remesh thos individual pieces into better shapes with less skewed elements.
156 might be better for a letter paper though

157 3.1.4. DEM input file settings

158 For the DEM simulations, a input script has to be prepared for LIGGGHTS. The input file
159 consists of various physical parameters of the particles being simulated, command lines for the
160 desired outputs from LIGGGHTS and various dump command which is used for post - processing
161 of the data.

162 Talk about mono and advantages and then psd. All the physical data is represented in the
163 Table 1

164 Need to add contact models, movement commands, all the start options,

165 3.1.5. DEM data post processing

166 The post processing of the data obtained from the DEM simulations was done using MATLAB®.
167 The first test run on the output data was to determine if the simulation had reached steady-state.
168 The mass inside the granulator was found out by averaging it over 5 time steps and then compared
169 to mass inside the granulator after every 10000 time steps (about 5×10^{-4} seconds) with a tolerance
170 of about 10

171 A precautionary script was also run so as to determine that no particles were lost due to overlap
172 of the geometry with the particles as well as from particle particle overlap.

173 3.2. PBM

174 3.2.1. Model Development

The main PBM equation developed for this work can be expressed as shown below:

$$\frac{d}{dt} F(s_1, s_2, x) = \mathfrak{R}_{agg}(s_1, s_2, x) + R_{break}(s_1, s_2, x) + \dot{F}_{in}(s_1, s_2, x) - \dot{F}_{out}(s_1, s_2, x) \quad (4)$$

175 citation?

176 where, $F(s_1, s_2, x)$ is the number of particles with an API volume of s_1 and an excipient volume
177 of s_2 in the spatial compartment x . The rate of change of number of particles with time in
178 different size classes depend on the rate of aggregation $\mathfrak{R}_{agg}(s_1, s_2, x)$ and the rate of breakage
179 $R_{break}(s_1, s_2, x)$. Also, the rate of particles coming into, $F_{in}(s_1, s_2, x)$ and going out, $F_{out}(s_1, s_2, x)$
180 of the spatial compartment due to particle transfer affect the number of particles in different size
181 classes. The rate of change of liquid volume is calculated using the equation:

[H]

Table 1: Physical Properties of the particle for the LIGGGHTS input script (Currently a placeholder)

Parameter	Symbol	Value	Units
Boltzmann constant	k	$1.3806488 \times 10^{-23}$	$m^2 kg s^{-2} K^{-1}$
Charge of electron	e	$1.60217657 \times 10^{-19}$	Coulombs
Avogadro Number	N_A	6.0221413×10^{23}	—
Hamaker constant	A_H	3×10^{-21}	J
Hydration force constant	F_0	10^{-2}	$N m^{-1}$
Temperature of the medium	T	298	K
Viscosity of the medium	μ	0.8999×10^{-3}	Pa.s
Permittivity of the medium	$\epsilon_0 \epsilon_r$	6.93×10^{-10}	$C^2 N^{-1} m^{-2}$
Valence of ions in medium	z	1	—
Bulk concentration of ions in medium	C^b	1×10^{-2}	$kg m^{-3}$
Debye length	$\frac{1}{\kappa}$	1.3581×10^{-7}	m
Decay length	δ_0	6×10^{-10}	m
Density of alginate	$\rho_{Alginate}$	1050	$kg m^{-3}$
Density of chitosan	$\rho_{Chitosan}$	1000	$kg m^{-3}$
Surface potential of alginate	$\Psi_{Alginate}$	-46×10^{-3}	Volts
Surface potential of chitosan	$\Psi_{Chitosan}$	56×10^{-3}	Volts
Volume of the system	V	10×10^{-6}	m^3
Volume of the smallest alginate bin	a_1	1.5×10^{-17}	m^3
Volume of the smallest chitosan bin	c_1	0.3×10^{-17}	m^3
Aggregation kernel constant	K_0	5×10^9	—
Simulated Process Time	t	10	s
Simulation Time-Step	dt	0.01	s

$$\begin{aligned} \frac{d}{dt} F(s_1, s_2, x) l(s_1, s_2, x) = & \mathfrak{R}_{liq,agg}(s_1, s_2, x) + \mathfrak{R}_{liq,break}(s_1, s_2, x) + \dot{F}_{in}(s_1, s_2, x) l_{in}(s_1, s_2, x) \\ & - \dot{F}_{out}(s_1, s_2, x) l_{out}(s_1, s_2, x) + F(s_1, s_2, x) \dot{l}_{add}(s_1, s_2, x) \end{aligned} \quad (5)$$

where, $l(s_1, s_2, x)$ is the amount of liquid volume in each particle with API volume of s_1 and excipient volume of s_2 in the spatial compartment x . $\mathfrak{R}_{liq,agg}(s_1, s_2, x)$ and $\mathfrak{R}_{liq,break}(s_1, s_2, x)$ are respectively the rates of liquid transferred between size classed due to aggregation and breakage. $l_{in}(s_1, s_2, x)$ and $l_{out}(s_1, s_2, x)$ are respectively the liquid volumes of the particles coming in and going out of the spatial compartment. $l_{add}(s_1, s_2, x)$ is the volume of liquid acquired by each particle in the compartment at every time step due to external liquid addition.

Similarly, the rate of change of gas volume is calculated using the following equation:

$$\begin{aligned} \frac{d}{dt} F(s_1, s_2, x) g(s_1, s_2, x) = & \mathfrak{R}_{gas,agg}(s_1, s_2, x) + \mathfrak{R}_{gas,break}(s_1, s_2, x) + \dot{F}_{in}(s_1, s_2, x) g_{in}(s_1, s_2, x) \\ & - \dot{F}_{out}(s_1, s_2, x) g_{out}(s_1, s_2, x) + F(s_1, s_2, x) \dot{g}_{cons}(s_1, s_2, x) \end{aligned} \quad (6)$$

citation?

where, $g(s_1, s_2, x)$ is the gas volume of each particle with API volume of s_1 and excipient volume of s_2 in the spatial compartment x . $\mathfrak{R}_{gas,agg}(s_1, s_2, x)$ and $\mathfrak{R}_{gas,break}(s_1, s_2, x)$ are respectively

191 the rates of gas transferred between size classed due to aggregation and breakage. $g_{in}(s_1, s_2, x)$
 192 and $g_{out}(s_1, s_2, x)$ are respectively the gas volume of the particles entering and leaving the spatial
 193 compartment. $g_{cons}(s_1, s_2, x)$ is the volume of gas coming out of each particle in the compartment
 194 at every time-step due to consolidation of the particles.

The rate of aggregation, $\mathfrak{R}_{agg}(s_1, s_2, x)$ in Equation 4 is calculated as

$$\begin{aligned}\mathfrak{R}_{agg}(s_1, s_2, x) = & \frac{1}{2} \int_0^{s_1} \int_0^{s_2} \beta(s'_1, s'_2, s_1 - s'_1, s_2 - s'_2, x) F(s'_1, s'_2, x) F(s_1 - s'_1, s_2 - s'_2, x) ds'_1 ds'_2 \\ & - F(s_1, s_2, x) \int_0^{s_{max1}-s_1} \int_0^{s_{max2}-s_2} \beta(s_1, s_2, s'_1, s'_2, x) F(s'_1, s'_2, x) ds'_1 ds'_2\end{aligned}\quad (7)$$

195 **citation?**

where, the aggregation kernel, $\beta(s_1, s_2, s'_1, s'_2, x)$ is expressed as

$$\begin{aligned}\beta(s_1, s_2, s'_1, s'_2, x) = & \beta_o * (V(s_1, s_2, x) + V(s'_1, s'_2, x))^\gamma * (c(s_1, s_2, x) \\ & + c(s'_1, s'_2, x))^\alpha \left(1 - \frac{(c(s_1, s_2, x) + c(s'_1, s'_2, x))^\delta}{2}\right)^\alpha\end{aligned}\quad (8)$$

196 **citation?**

197 where, β_o , α , δ and γ are aggregation rate constants, $V(s_1, s_2, x)$ and $V(s'_1, s'_2, x)$ are the
 198 volumes of the aggregating particles. $c(s_1, s_2, x)$ and $c(s'_1, s'_2, x)$ are the external liquid fraction of
 199 the aggregating particles.

200 Similarly, the breakage rate is expressed as-

$$\mathfrak{R}_{break}(s_1, s_2, x) = \int_0^{s_{max1}} \int_0^{s_{max2}} K_{break}(s'_1, s'_2, x) F(s'_1, s'_2, x) ds'_1 ds'_2 - K_{break}(s_1, s_2, x) F(s_1, s_2, x)\quad (9)$$

201 **citation?**

202 where, the breakage kernel $K_{break}(s_1, s_2, x)$ is formulated as

$$K_{break}(s_1, s_2, x) = \left(\frac{4}{15\pi}\right)^{\left(\frac{1}{2}\right)} G_{shear} \exp\left(-\frac{B}{R(s_1, s_2, x)}\right)\quad (10)$$

203 **citation?**

204 where, G_{shear} is the shear rate exerted by the impeller on the granules. $R(s_1, s_2, x)$ is the
 205 radius of the granule that breaks and B is the breakage kernel constant. G_{shear} is calculated as
 206 $\frac{\nu_{impeller}*D_{impeller}*PI}{60}$ where $\nu_{impeller}$ and $D_{impeller}$ are respectively the rotational speed and diameter
 207 of the impeller.

208 The rate of increase of liquid volume of one particle, $\dot{l}_{add}(s_1, s_2, x)$ is expressed as $\frac{(s_1+s_2)(\dot{m}_{spray}(1-c_{binder})-\dot{m}_{evap})}{m_{solid}(x)}$
 209 where, $(s_1 + s_2)$ is the total solid volume of the particle; \dot{m}_{spray} is the rate of external liquid ad-
 210 dition, c_{binder} is the concentration of binder in the external liquid (which is assumed to be zero
 211 in this case as pure liquid is added); \dot{m}_{evap} is the rate of evaporation of liquid from the system
 212 (which is also assumed to be zero in this case) and m_{solid} is the total amount of solid present in
 213 the compartment.

214 The rate of decrease in gas volume per particle due to consolidation is calculated using the
 215 following expression:

$$\dot{g}_{cons}(s_1, s_2, x) = c * (\nu_{impeller})^\omega * V(s_1, s_2, x) \frac{(1 - \epsilon_{min})}{s} [g(s_1, s_2, x) + l(s_1, s_2, x) - (s_1 + s_2) \frac{\epsilon_{min}}{1 - \epsilon_{min}}] \quad (11)$$

where, c and ω are the consolidation constants; $\nu_{impeller}$ is the impeller rotational speed; $V(s_1, s_2, x)$ is the volume of particle, ϵ_{min} is the minimum porosity; $g(s_1, s_2, x)$ and $l(s_1, s_2, x)$ are respectively the gas and liquid volume of the particle.

Particle transfer rate, $F_{out}(s_1, s_2, x)$ in Equation 4 is calculated as $F(s_1, s_2, x) * \frac{\nu_{compartment(x)} * dt}{d_{compartment}}$ where, $\nu_{compartment(x)}$ and $d_{compartment}$ are respectively the average velocity of particles in compartment x and the distance between the mid-points of two adjacent compartment, which is the distance particles have to travel to move to the next spatial compartment. dt is the time-step. The values of various parameters used in the model are provided in Table 2.

Table 2: Parameters for PBM from Anik's hetero. agg. paper. currently place holder

Parameter	Symbol	Value	Units
Boltzmann constant	k	$1.3806488 \times 10^{-23}$	$m^2 \text{ kg s}^{-2} K^{-1}$
Charge of electron	e	$1.60217657 \times 10^{-19}$	Coulombs
Avogadro Number	N_A	6.0221413×10^{23}	—
Hamaker constant	A_H	3×10^{-21}	J
Hydration force constant	F_0	10^{-2}	$N m^{-1}$
Temperature of the medium	T	298	K
Viscosity of the medium	μ	0.8999×10^{-3}	Pa.s
Permittivity of the medium	$\epsilon_0 \epsilon_r$	6.93×10^{-10}	$C^2 N^{-1} m^{-2}$
Valence of ions in medium	z	1	—
Bulk concentration of ions in medium	C^b	1×10^{-2}	$kg m^{-3}$
Debye length	$\frac{1}{\kappa}$	1.3581×10^{-7}	m
Decay length	δ_0	6×10^{-10}	m
Density of alginate	$\rho_{Alginate}$	1050	$kg m^{-3}$
Density of chitosan	$\rho_{Chitosan}$	1000	$kg m^{-3}$
Surface potential of alginate	$\Psi_{Alginate}$	-46×10^{-3}	Volts
Surface potential of chitosan	$\Psi_{Chitosan}$	56×10^{-3}	Volts
Volume of the system	V	10×10^{-6}	m^3
Volume of the smallest alginate bin	a_1	1.5×10^{-17}	m^3
Volume of the smallest chitosan bin	c_1	0.3×10^{-17}	m^3
Aggregation kernel constant	K_0	5×10^9	—
Simulated Process Time	t	10	s
Simulation Time-Step	dt	0.01	s

3.2.2. Parameters

3.3. PBM Parallel C++

3.3.1. Discretization & Parallelizing PBM

To solve the PBM numerically it was discretized using a finite element method type solution. To obtain the most optimal parallel performance, when solving the PBM, work loads were distributed

229 in a manner which took into account the shared memory and distributed memory aspects of the
230 clusters the PBM was being run on. To parallelize the model in a way which could use shared and
231 distributed methods of OMP and MPI were used.

232 One MPI process was used per CPU socket and one OMP process was used per CPU core, as
233 authors (Bettencourt et al. (2017)) found it resulted in the best performance. MPI was used for
234 message passing from one node to another while OMP was used for calculations on each node that
235 could be efficiently solved using a shared memory system i.e. calculations were inter-dependent but
236 could be computed simultaneously.

237 Pseudo code is presented below illustrating how the calculations are distributed and carried out
238 during the simulation. For each time step the MPI processes are made responsible for a specific
239 chunk of the spatial compartments. Then each OMP thread, inside of each MPI process, is allocated
240 to one of the cores of the multi-core CPU the MPI process is bound too. The OMP processes
241 divide up and compute \mathfrak{R}_{agg} and \mathfrak{R}_{brk} . (include more detail about how they do it? last paper
242 reviewer complained that could not understand figure by JUST reading what I wrote about it in
243 meat of paper)

244 After \mathfrak{R}_{agg} and \mathfrak{R}_{brk} are calculated the MPI processes calculate the new PSD value for their
245 chunk at that specific time step, $F_{t,c}$. The slave processes send their $F_{t,c}$ to the master processes
246 which collects them into the complete $F_{t,all}$. The master process then broadcasts the $F_{t,all}$ value to
247 all slave processes.

248 A crucial feature of the PBM is that the current PSD ($F_{t,all}$) value is used to compute a new
249 time step size for the next iteration. This means all of the MPI processes need to have the same
250 dynamic time step size at each iteration for the calculations to be properly carried out in parallel.
251 Since the completely updated $F_{t,all}$ value is shared before calculating a new time step each process
252 will have the same $F_{t,all}$ value. As a result each process calculates the same size for the new time
253 step. Did not include the liquid and gas PBMs in this but hoping they will be some what assumed?
254 Also the Ragg omp distributed work is an a what about private OMP vars specified that has impact
255 on how model is solved etc. Should look into this. might change based on locking/blocking tests
256 that need to be implemented still.

Algorithm 1 Pseudo code

```

while  $t < t_{final}$  do
    // the spatial domain is divided into equal chunks (with in 1 bin size)
    // each MPI process is assigned on chunk of spatial domain shown as  $c_{low}$  to  $c_{up}$ 
    // sum all  $c_{low_i}$  to  $c_{up_i}$  is = to [0,numCompartments]
    for each MPI processes do  $c = c_{low_i}$  to  $c_{up_i}$ 
        // each MPI process is further divided with OMP to take advantage of multi-core CPU
        // each OMP process is allocated to a single compute core
        //  $\Re$  integrals (i1)  $\int_0^{s_2}$ , (i2)  $\int_0^{s_{max2}-s_2}$ , and (i3)  $\int_0^{s_{max2}-s_2}$  are divided into smaller integrals
        //  $\int_{i_1 low_n}^{i_1 up_n}$ ,  $\int_{i_2 low_n}^{i_2 up_n}$ , and  $\int_{i_3 low_n}^{i_3 up_n}$  which are solved by the "n" OMP processes
        // allocated to that MPI process (CPU)
        for each OMP process do

            
$$\Re_{agg}(s_1, s_2, c) = \frac{1}{2} \int_0^{s_1} \int_{i_1 low_n}^{i_1 up_n} \beta(s'_1, s'_2, s_1 - s'_1, s_2 - s'_2, c) F(s'_1, s'_2, c) F(s_1 - s'_1, s_2 - s'_2, c) ds'_1 ds'_2$$

            
$$- F(s_1, s_2, c) \int_0^{s_{max1}-s_1} \int_{i_2 low_n}^{i_2 up_n} \beta(s_1, s_2, s'_1, s'_2, c) F(s'_1, s'_2, c) ds'_1 ds'_2$$


            
$$\Re_{break}(s_1, s_2, c) = \int_0^{s_{max1}} \int_{i_3 low_n}^{i_3 up_n} K_{break}(s'_1, s'_2, c) F(s'_1, s'_2, c) ds'_1 ds'_2 - K_{break}(s_1, s_2, c) F(s_1, s_2, c)$$


        end for

        
$$F_{t,c} = \frac{\Delta F(s_1, s_2, c)}{\Delta t} \Delta t + F(s_1, s_2, c)_{t-1}$$

        
$$= (\Re_{agg}(s_1, s_2, c) + \Re_{break}(s_1, s_2, c) + \dot{F}_{in}(s_1, s_2, c) - \dot{F}_{out}(s_1, s_2, c)) \Delta t + F(s_1, s_2, c)_{t-1}$$


    end for
    MPI Send  $F_{t,c}$  to Master MPI process
    MPI Recv  $F_{t,c}$  from MPI all slave processes
    Master consolidate all  $F_{t,c}$  chunks into a complete  $F_{t,all}$ 
    Master does inter-bin particle transfers (updates  $F_{t,all}$ )
    MPI Broadcast  $F_{t,all}$  to all slave processes
     $t_{new} = t + timestep$ 
end while

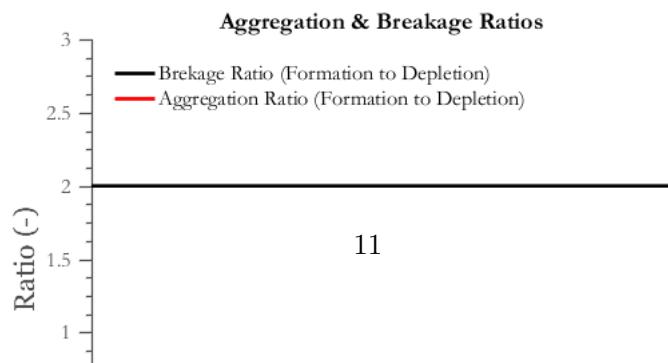
```

257 3.4. RP & PBM+DEM Communication

258 **4. Results**

259 4.1. PBM

260 4.1.1. PBM Validation



261 The ratio between the number of particles formed due to aggregation and the number of particles
262 depleted due to aggregation and the ratio of the number of particles formed due to breakage to the
263 number of particles depleted due to breakage are plotted. In aggregation two particles agglomerate
264 to form one particle and in breakage one particle breaks to form two particles. So, these ratios
265 are expected to be 0.5 and 2 respectively. As can be seen from Figure 4, these ratios are accurate
266 confirming that mass is conserved accurately in the model.

267 The granulator was divided into 3 compartments spatially and the total volume, solid volume
268 and pore volume and the median diameter d_{50} in each compartment were plotted to study the
269 granulation behaviour and are shown in Figure 5.

270 It can be seen from Figure 5a that the total volume starts to increase first in compartment 1
271 followed by compartment 2 and then compartment 3. This happens as gradually particles entering
272 compartment 1 moves to the other compartment due to particle transfer from compartment 1 to
273 compartment 2 and then compartment 3. In Figure 5b it is observed that the solid volume similar
274 to the total volume increases first in compartment 1 and last in compartment 3. The solid volume
275 becomes constant and equal in all the compartments at around 30-50 seconds and steady state is
276 reached when the rate of particle volume being transported through the compartments and leaving
277 the system is equal to the rate of particles entering the system. Although, as seen in Figure 5c
278 the pore volume which is the sum of the gas and the liquid volume is highest in compartment 3
279 and lowest in compartment 1. This happens due to the external liquid addition to the system. As
280 the particles move from compartment 1 to compartment 3, they gradually acquire a higher amount
281 liquid, thereby increasing the pore volume. In Figure 5d, the D_{50} is seen to be increasing from
282 compartment 1 to 3. This happens because of the size enlargement of large particles coming in
283 from the previous compartment because of the external liquid added to each compartment and a
284 longer residence time in the granulator.

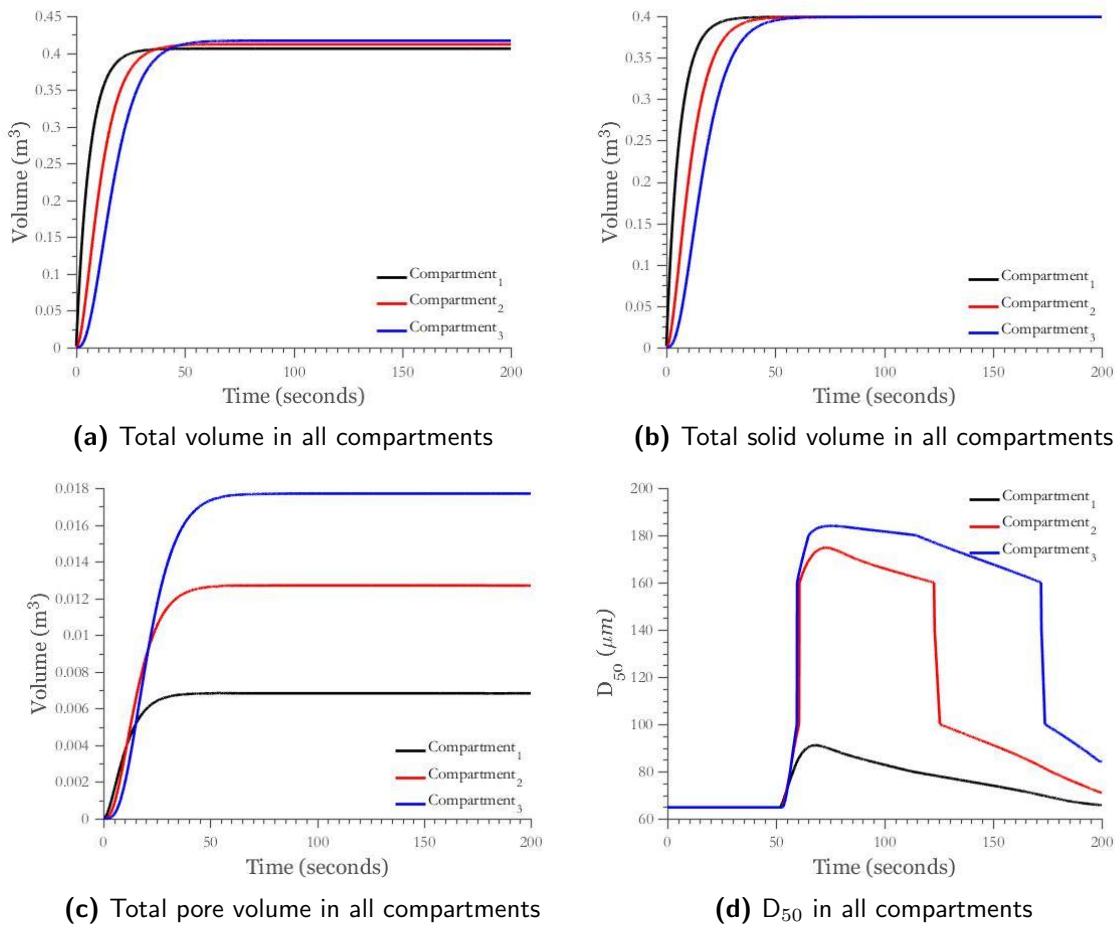


Figure 5: Volume and D_{50} in all compartments over time. Volumes become constant as steady state is reached. Median diameter increases and then decreases as bigger particles leave the system and smaller particles occupy that volume.

285 4.1.2. Parallel C++ PBM Validation

286 show PSD or D_{50} is the same as Matlab or serial PBM

287 1. fig D_{50} Matlab vs Parallel

288 4.1.3. Parallel PBM Performance

289 show that RP has minimal impact on performance

290 show that performance is mostly unaffected by RP

291 1. fig scaling

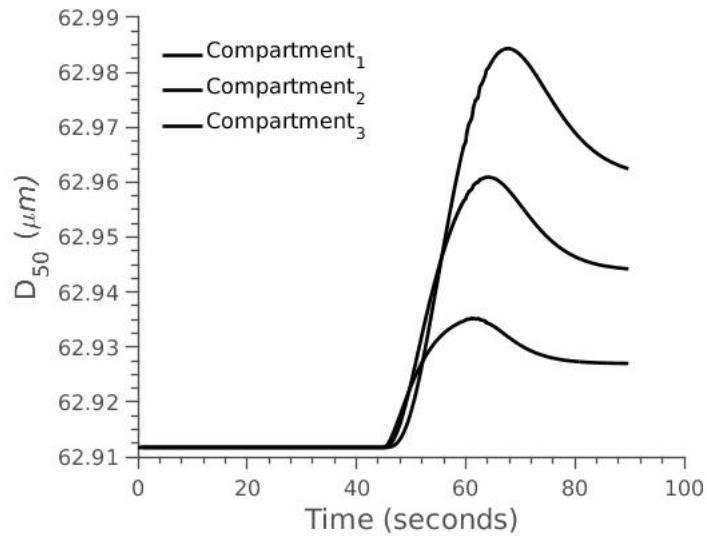


Figure 6: D_{50} of Matlab PBM vs Parallel PBM

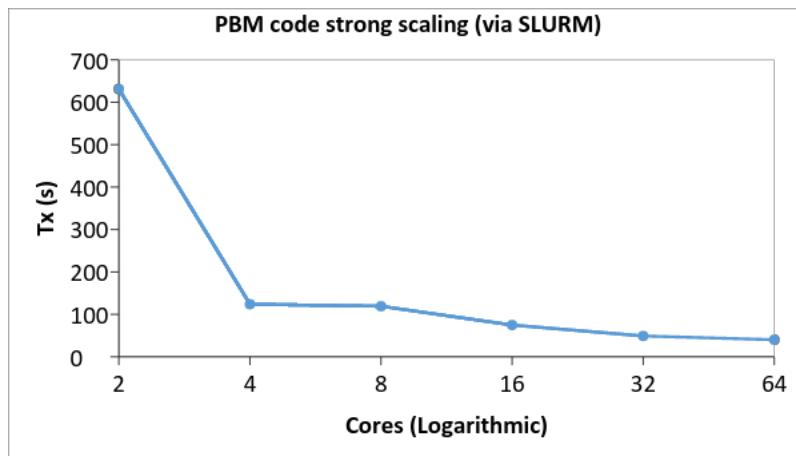


Figure 7: PBM strong scale slurm

292 2. fig scaling w/ RP

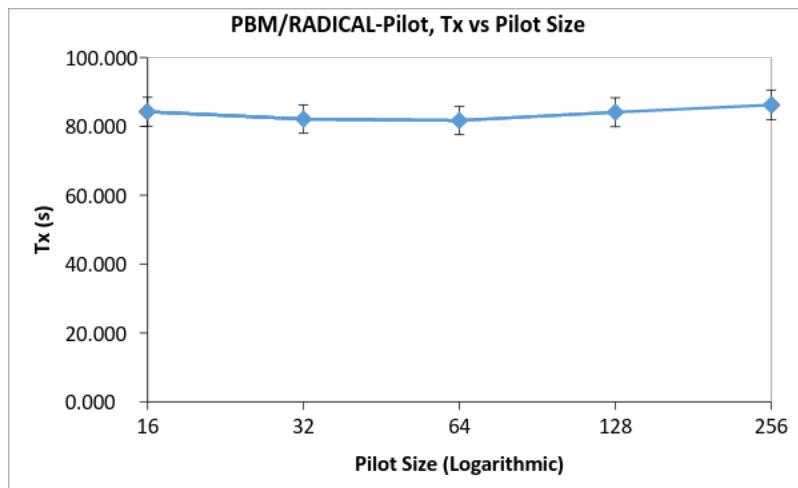


Figure 8: PBM strong scale RP

293 4.1.4. Parallel PBM Parameter space and Parameter Estimation

- 294 I. show how effective parallel pbm is for parameter estimation
- 295 II. Find/ explore ranges of DEM data that PBM can use to find Critical parameters and
- 296 sensitivities will be useful to us in linking and in picking best DEM parameters to vary and best
- 297 parameters for PBM+DEM code
 - 298 1. fig range of some parameters?
 - 299 2. fig range of other pbm parameter ?

300 4.2. DEM

301 4.2.1. DEM Validation and Parameter Space Studies

- 302 show that DEM is somewhat behaving like a real system
- 303 fig constant flow reached by end of DEM simulation

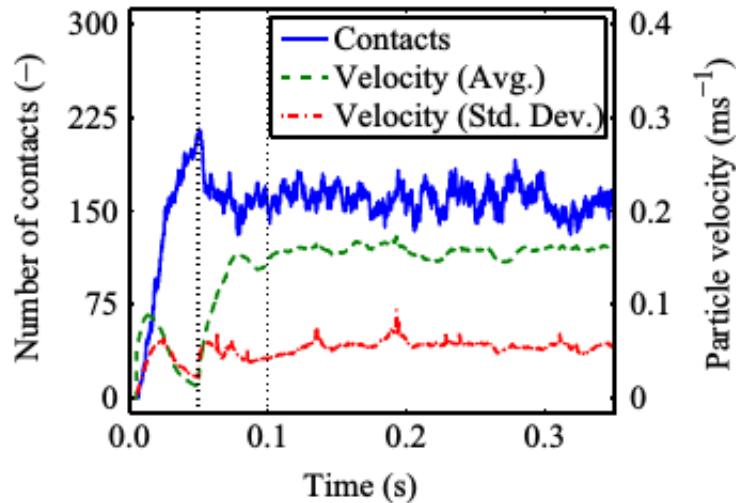


Fig. 3. Time trajectories from DEM simulation showing number of contacts between 1-mm particles with particles of the same size, and the average and standard deviation of the velocity of 1-mm particles. Particles are added at time=0 s with a median diameter of 1.5 mm and a standard deviation of 0.5 mm. Dotted lines indicate start of rotation of the screw and start of data collection, respectively.

Figure 9

304 fig RTD
 305 show affects of certain parameters on DEM and which ones are most critical to outcomes - will
 306 help decide on PBM+DEM parameters to study in later section
 307 fig vary impeller RPM see how RTD or hold up changes
 308 fig vary PSD (range and/or particle sizes) to see how C_{coll} etc will be affected - important for
 309 PBM Kernel

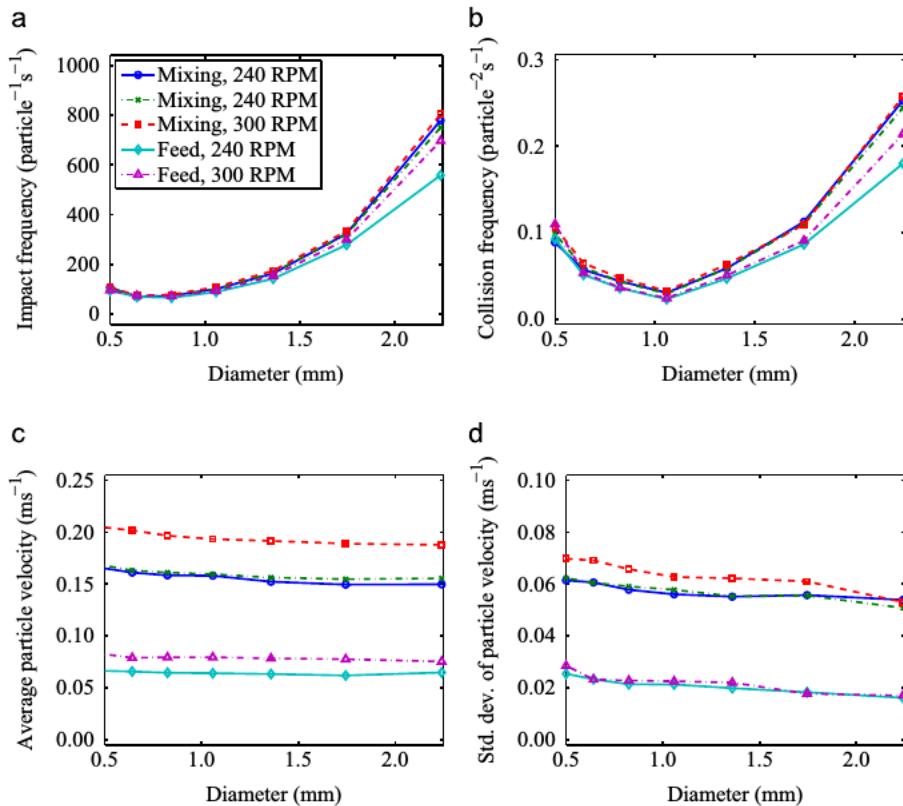


Fig. 6. DEM results showing the effects of screw element type and rotational speed on (a) impact frequency, (b) collision frequency with 1-mm particles, (c) average particle velocity, and (d) standard deviation of particle velocity for each size class.

Figure 10: fig showing sensitivity of C_{coll} and etc to RPM

310 fig vary impeller RPM see C_{coll} changes

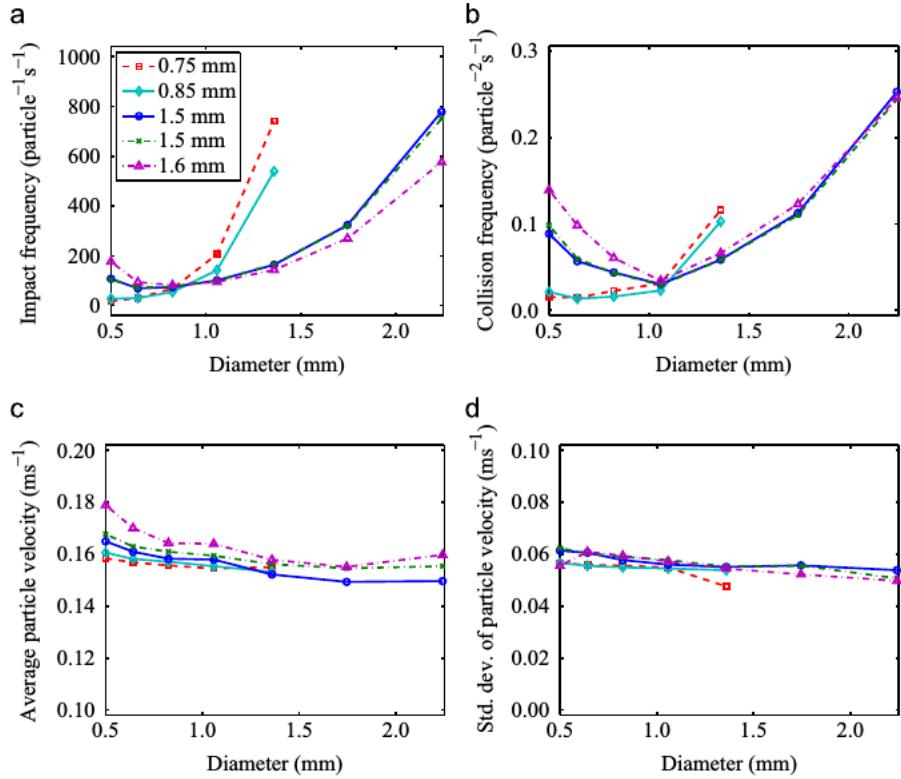


Fig. 4. DEM results showing the effects of size distribution on (a) impact frequency, (b) collision frequency with 1-mm particles, (c) average particle velocity, and (d) standard deviation of particle velocity for each size class. Median diameters are listed in the legend. The standard deviation of the diameter is fixed at 0.2 mm for the smallest two simulations and 0.5 mm for the larger simulations.

Figure 11: fig from dana 2015 mechanistic bi-directional

311 4.2.2. DEM Performance

312 show that performance is mostly unaffected by RP
 313 1. fig scaling

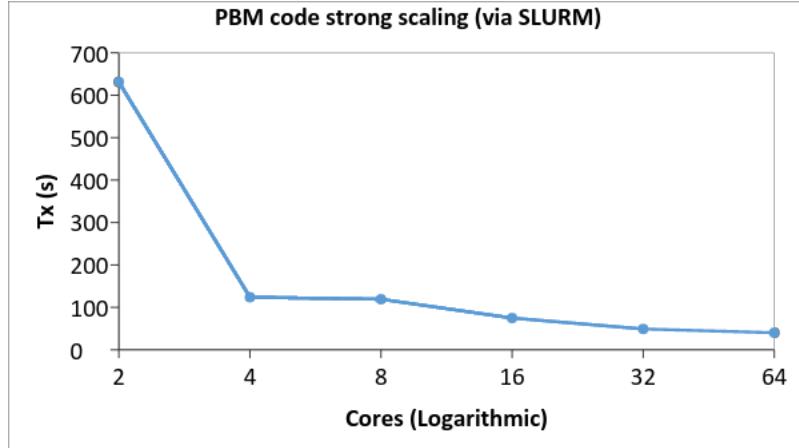


Figure 12: DEM scale just slurm

314 2. fig scaling w/ RP

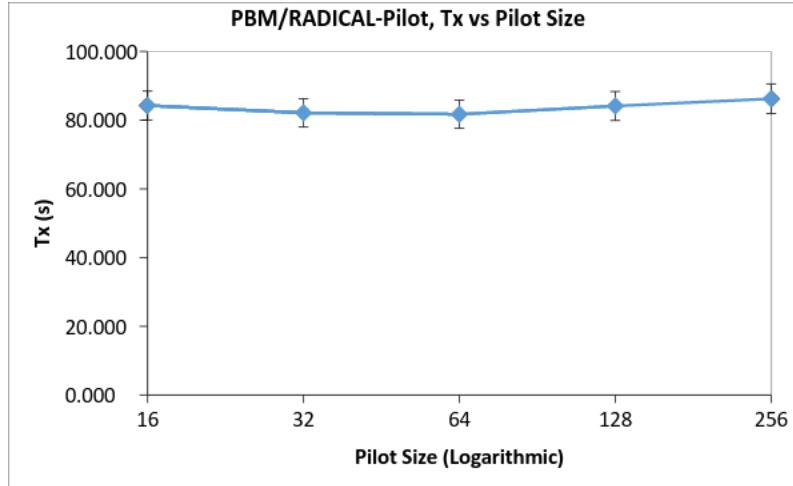


Figure 13: DEM scale with RP

315 4.3. PBM+DEM - RP

316 4.3.1. PBM+DEM Validation/Accuracy?

317 4.3.2. PBM+DEM Performance

318 strong scaling

fig PBM + DEM RP strong scaling

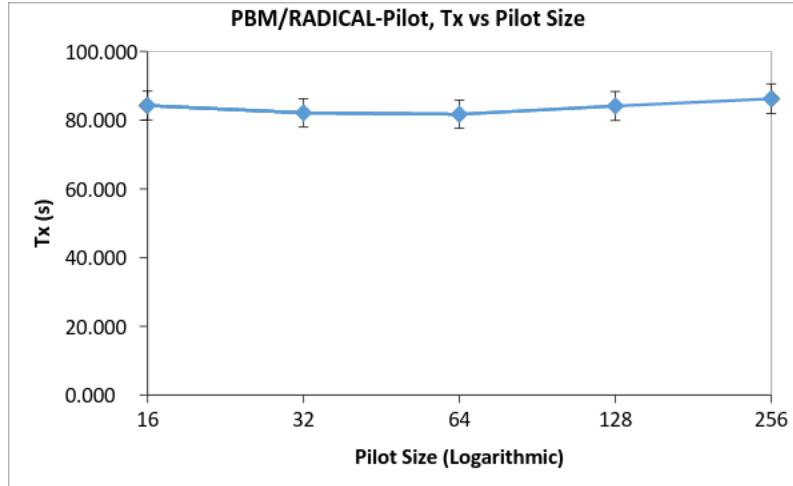


Figure 14: PBM+DEM scale with RP

319

320 4.3.3. PBM+DEM Parameter studies

321 show how PBM+DEM captures multi-physics as parameters changed. helps validate and support model development. show we have made a useful tool for future work.

323 fig

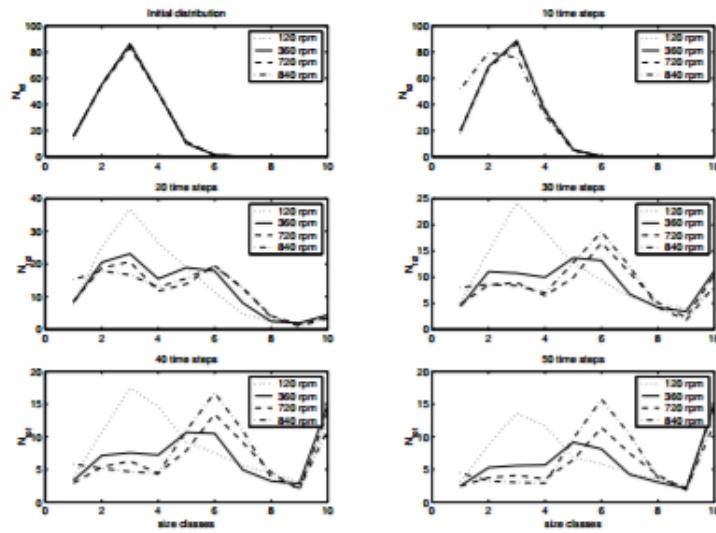


Figure 8: PSD evolution for simulations run at 120 rpm, 360 rpm, 720 and 840 rpm.

Figure 15: PBM+DEM scale with RP

324 PSD over time for different impeller RPM

325 **5. Conclusions**

326 **References**

- 327 Adhianto, L., Chapman, B., 2007. Performance modeling of communication and computation in
328 hybrid {MPI} and openmp applications. Simulation Modelling Practice and Theory 15 (4), 481
329 – 491, performance Modelling and Analysis of Communication Systems.
330 URL <http://www.sciencedirect.com/science/article/pii/S1569190X06001109>
- 331 Barraso, D., Ramachandran, R., 2015. Multi-scale modeling of granulation processes: Bi-
332 directional coupling of {PBM} with {DEM} via collision frequencies. Chemical Engineering Re-
333 search and Design 93, 304 – 317.
334 URL <http://www.sciencedirect.com/science/article/pii/S0263876214001920>
- 335 Barraso, D., Walia, S., Ramachandran, R., 2013. Multi-component population balance modeling of
336 continuous granulation processes: A parametric study and comparison with experimental trends.
337 Powder Technology 241, 85 – 97.
338 URL <http://www.sciencedirect.com/science/article/pii/S0032591013001666>
- 339 Bettencourt, F. E., Chaturbedi, A., Ramachandran, R., 2017. Parallelization methods for efficient
340 simulation of high dimensional population balance models of granulation. Computers Chemical
341 Engineering, –.
342 URL <http://www.sciencedirect.com/science/article/pii/S0098135417301072>
- 343 Cundall, P. A., Strack, O. D. L., 1979. A discrete numerical model for granular assemblies. Gotech-
344 nique 29 (1), 47–65.
345 URL <http://dx.doi.org/10.1680/geot.1979.29.1.47>

- 346 Jin, H., Jespersen, D., Mehrotra, P., Biswas, R., Huang, L., Chapman, B., 2011. High performance
347 computing using {MPI} and openmp on multi-core parallel systems. Parallel Computing 37 (9),
348 562 – 575, emerging Programming Paradigms for Large-Scale Scientific Computing.
349 URL <http://www.sciencedirect.com/science/article/pii/S0167819111000159>
- 350 Kloss, C., Goniva, C., Hager, A., Amberger, S., Pirker, S., 2012. Models, algorithms and validation
351 for opensource dem and cfd-dem. Progress in Computational Fluid Dynamics, an International
352 Journal 12 (2-3), 140–152.
- 353 Ramkrishna, D., Singh, M. R., 2014. Population Balance Modeling: Current Status and Future
354 Prospects. Annual Review of Chemical and Biomolecular Engineering 5 (1), 123–146.
355 URL <http://dx.doi.org/10.1146/annurev-chembioeng-060713-040241>
- 356 Wilkinson, B., Allen, C., 2005. Parallel Programming: Techniques and Applications Using Net-
357 worked Workstations and Parallel Computers. An Alan R. Apt book. Pearson/Prentice Hall.
358 URL <https://books.google.com/books?id=F8C2QgAACAAJ>