

FINAL SCIENTIFIC/TECHNICAL REPORT for DE-FE0026393

2018-001

Unlimited Release

Printed May, 2018

(AOI 4B) MFIX-DEM Phi: Performance and Capability Improvements Towards Industrial Grade Open-source DEM Framework with Integrated Uncertainty Quantification

Aytakin Gel (SCIDSE-ASU), Yang Jiao (SEMTE-ASU),
Heather Emady (SEMTE-ASU), Charles Tong (LLNL)

Prepared by
Arizona State University
Tempe, Arizona 85281



Issued by Arizona State University

DISCLAIMER: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

2018-001
Unlimited Release
Printed May, 2018

(AOI 4B) MFIx-DEM Phi: Performance and Capability
Improvements
Towards Industrial Grade Open-source DEM Framework
with Integrated Uncertainty Quantification

Aytekin Gel, Yang Jiao, Heather Emady, Charles Tong

This page is intentionally left blank.

Contents

Acknowledgment	13
Executive Summary	14
Technical Summary	15
Task 1 - Project Management and Planning	18
Task 2 - Performance Improvements in MFIX-DEM	19
Prior Performance Profiling Results Without Polydispersity	20
Most Recent Performance Profiling Results With Polydispersity	27
Task 3 - Physical Modeling Capability Enhancements in MFIX-DEM	29
Subtask 3.1 - Enhance the Capability for Handling Particle Size Distributions	29
3.1.1 - Implementation of Polydispersity Features in MFIX-DEM	29
3.1.2 - Verification and Validation (V&V) Using Hopper Bin Discharge Studies	30
3.1.3 - Particle Height Classification Problem	42
Subtask 3.2 - Validation of Heat Transfer Models	57
3.2.1 - MFIX-DEM Heat Transfer Verification and Validation	57
Task 4 - Ascertain Predictive Credibility in MFIX-DEM	69
Task 5 - Perform Industry Outreach and Development of Industry Cases	77
Conclusions & Future Work Recommendations	78
References	81

List of Figures

1	Overview of the Funding Opportunity Announcement (FOA 0001238) objectives and associated subtasks implemented in the project to address these objectives	15
2	Schematic depiction of the contributions proposed in this project, original illustration adapted from Musser et al. [23]	16
3	Listing of the Funding Opportunity Announcement (FOA 0001238) objectives and associated subtasks implemented to address these objectives	16
4	Dashboard screen showing the performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.	21
5	Intel Advisor tool based performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) showing the top loops that consume most of the time during the execution session.	21
6	Intel Advisor screen displaying the location of the top bottleneck loop in the source code with some execution metrics such as loop time.	22
7	Display of the second top time-consuming loop during the execution of Benchmark Case 1 on Intel Xeon Haswell (HSW)	22
8	Compiler generated assembly language view of the second top time-consuming loop determined by Intel Advisor.	23
9	Performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.	23
10	Summary display for performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the Intel VTune tool to analyze general performance bottlenecks.	24
11	Performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the Intel VTune Amplifier tool showing the most time consuming subroutines.	25
12	Detailed view of the hot-spots in the top time-consuming subroutine <code>DESGRID_NEIGH_BUILD</code> for Benchmark Case 2 as identified by Intel VTune Amplifier.	25

13	Summary report for performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.	26
14	Detailed view of the top time-consuming loops determined during performance profiling of Benchmark Case 2 with Intel Advisor.	26
15	Revised benchmark problem	27
16	Intel Advisor Profiling of Polydispersity implemented 2016-1 version of MFIX-DEM	27
17	Detail view of Intel Advisor Profiling of Polydispersity implemented 2016-1 version of MFIX-DEM	28
18	Discharge dynamics for a 3D hopper with equal-sized spherical beads. The discharged mass vs. discharge time curves obtained from both the 2016-1 MFIX and our new implementation agree well with one another.	31
19	Experimental setup of hopper: Clamped 3D printed hopper with a beaker, for collecting discharged particles, placed on an analytical balance to measure the mass of discharged particles for sampling.	32
20	A bi-modal particle size distribution (i.e., number frequency vs. particle sizes) was employed, which includes two normal distributions for fine and coarse silica bead particles.	34
21	Silica beads: Normal distribution of the fine particles with mean of 1.5 mm and standard deviation of 0.3 mm, colored in red; and the coarse particles with a mean of 2.9 mm and standard deviation of 0.1 mm, colored in green. .	35
22	Particle arrangement: (a) First discharged sample from the well-mixed configuration. An approximately equal number of fine and coarse particles are discharged. (b) Last discharged sample from the well-mixed configuration. A majority of fine particles are discharged. (c) First discharged sample from the layered configuration. Only coarse particles are discharged. (d) Last discharged sample from the layered configuration. A mixture of both fine and coarse particles are discharged.	36
23	Initial configuration of the discharge hopper containing a well-mixed packing of spherical beads with a bi-modal size distribution. The unit of the particle diameter is centimeter.	38
24	Snapshots of the discharge simulation of the well-mixed configuration at 6 s, 9 s, 11 s and 12 s, respectively, from left to right. The upper panels show the top view and the lower panels show the side view.	39

25	Discharge dynamics are quantified using the normalized fine mass fraction γ_f^N of the discharged particles vs. the overall discharged mass fraction $\bar{\gamma}$ for the well-mixed configuration. The experimental results are represented as a box plot.	39
26	Initial configuration of the hopper discharge containing a layered packing of spherical beads with a bi-modal size distribution. The unit of the particle diameter is centimeter.	40
27	Snapshots of the discharge simulation of the layered configuration at 6 s, 9 s, 11 s and 12 s, respectively, from left to right. The upper panels show the top view and the lower panels show the side view.	41
28	Discharge dynamics are quantified using the normalized fine mass fraction γ_f^N of the discharged particles vs. the overall discharged mass fraction $\bar{\gamma}$ for the layered configuration. The experimental results are represented as a box plot.	42
29	Schematic illustration of Convolutional Neural Network (CNN) model used for particle height classification.	43
30	Schematic illustration of four distinct initial packing geometries for two solid phases in a hopper bin. The blue region represents the denser phase (i.e., phase # 2) and the gray region represents phase # 1.	45
31	Scatter plots of resident time vs. particle height in hopper bin for the sphere geometries. The radii of the spherical region containing phase # 2 particles are respectively $R = 3.5, 4.0, 4.5$ and 5.0 cm.	45
32	Scatter plots of resident time vs. particle height in hopper bin for the cuboid geometries. The edge lengths of the square basal plane of cuboid containing phase # 2 particles are respectively $L = 3.0, 4.0, 5.0$ and 6.0 cm. The height of the cuboidal region is fixed with $H = 12.0$ cm.	46
33	Scatter plots of resident time vs. particle height in hopper bin for the upward cone geometries. The radii of the basal plane of cone containing phase # 2 particles are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm. The height of the cuboidal region is fixed with $H = 10.0$ cm.	47
34	Scatter plots of resident time vs. particle height in hopper bin for the downward cone geometries. The radii of the basal plane of cone containing phase # 2 particles are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm. The height of the cuboidal region is fixed with $H = 10.0$ cm.	48
35	Upper panels: Snapshots of the initial packing configuration for the sphere geometry. The radii of the spherical region are respectively $R = 3.5, 4.0, 4.5$ and 5.0 cm from left to right. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.	50

36	Upper panels: Snapshots of the initial packing configuration for the cuboid geometry. The edge lengths of the square basal plane are respectively $L = 3.0, 4.0, 5.0$ and 6.0 cm from left to right. The height of the cuboidal region is fixed with $H = 12.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.	50
37	Upper panels: Snapshots of the initial packing configuration for the upward cone geometry. The radii of the basal plane are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm from left to right. The height of the cone is fixed with $H = 10.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.	51
38	Upper panels: Snapshots of the initial packing configuration for the downward cone geometry. The radii of the basal plane are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm from left to right. The height of the cone is fixed with $H = 10.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.	52
39	The reconstruction accuracy M as a function of the tolerance σ for different bin width s for the geometry-specific (GS) models and the generic model (GE), compared to the corresponding hopper-specific (HS) models. One packing configuration not used for training the models is chosen from each geometry class. The results shown respectively correspond to the sphere, cuboid, upward cone and downward cone geometries.	53
40	Snapshot of the initial packing configuration for the four-phase system in a layered configuration. The particles for phases #1, 2, 3 and 4 are respectively shown with dark blue, light blue, green and red colors.	54
41	Scatter plots of resident time vs. particle height for each phases in the four-phase system. The data for phases #1, 2, 3 and 4 are respectively shown with dark blue, light blue, green and red colors.	55
42	The reconstruction accuracy M as a function of the tolerance σ for different bin width s for each phase in the four-phase system. The results for phases #1, 2, 3 and 4 are respectively shown.	56
43	Temperature distribution of monodispersed alumina obtained from MFI-DEM simulations at (a) 1 s, (b) 5 s, (c) 10 s, and (d) 20 s.	58
44	Temperature distribution of monodispersed alumina obtained from MFI-DEM simulations at (a) 1 s, (b) 5 s, (c) 10 s, and (d) 20 s.	59
45	The evolution of average bed temperature for the MFI-DEM simulation and Chaudhuri et al. (2010) DEM simulation.	59

46	Logarithmic variation of scaled temperature difference with time for the MFIX-DEM simulation and Chaudhuri et al. [6] DEM simulation.	60
47	The evolution of the scaled average bed temperature for the MFIX-DEM simulation and Chaudhuri et al. [6] experiment.	61
48	Logarithmic variation of scaled temperature difference with time for the MFIX-DEM simulation and Chaudhuri et al. [6] experiment.	61
49	The sensitivity of the overall heat transfer coefficient to rotation speed for the MFIX-DEM simulations and Emady et al. (2016) DEM simulations.	64
50	Temperature distribution of polydispersed alumina, obtained from MFIX-DEM simulations at $t=\tau/2$ for rotation speeds of (a) 5 rpm, (b) 10 rpm, (c) 20 rpm, and (d) 30 rpm.	64
51	Particle size distributions with $\mu = 0.4$ cm and $\sigma = 0.035$ cm.	65
52	The sensitivity of the overall heat transfer coefficient to rotation speed for three PSDs with $\mu = 0.4$ cm and $\sigma = 0.035$ cm.	65
53	Particle size distributions obtained from MFIX-DEM simulations at $t=\tau/2$ for (a) homogeneous, (b) narrow, and (c) broad distributions.	66
54	The sensitivity of the overall heat transfer coefficient to rotation speed at rolling frictions of 0, 0.005, 0.01, and 0.1.	67
55	Local experimental capabilities for rotary drum heat transfer for MFIX-DEM validation.	68
56	Depiction of an overview of the UQ capabilities and workflow embedded behind the GUI for MFIX-DEM (adapted from Gel et al.[14])	70
57	Illustration of the workflow proposed for UQ framework and proof-of-concept prototype implementation for Phase 1.	71
58	Illustration of the operation of the proof-of-concept prototype implemented for Phase 1: Step 1 user selects the UQ question that the answer is being sought.	72
59	Step 2: UQ GUI sets up the nodes necessary to complete the UQ analysis automatically within Nodeworks framework based on the UQ question selected.	72
60	Step 3: User enters few required information such as the variables to be considered as uncertain, the lower and upper bounds, number of sampling simulations that can be afforded, etc.	73
61	Step 4: User triggers the workflow by clicking the play button to run UQ GUI workflow, which in return interacts with PSUADE UQ toolbox behind the scenes. .	73

62	Step 5: Work directories to run independent MFI-X-DEM job for this specific case is generated by creating 12 sampling runs where coefficient of friction for particle to particle and particle to wall are varied using Monte Carlo simulation.	74
63	Illustration of the UQ wizard within Nodeworks with PSUADE integration.	75
64	Illustration of the nodes populated within the workspace based on the user's selection from UQ wizard within Nodeworks with PSUADE integration.	75
65	Illustration of the new UQ nodes added by porting from the original proof-of-concept demonstration within MFI-X-GUI into Nodeworks with PSUADE integration.	76

List of Tables

1	Verification simulation details for the discharge hopper with monodisperse spherical beads.	31
2	Geometrical parameters, masses of the two solid particle phases and global parameters used in validation simulations.	37
3	Simulation parameters used for verifying the effect of rotation speed on the contact conduction model in MFI-X-DEM with polydispersed alumina particles.	63
4	Particle size distribution information used for the narrow, homogeneous, and broad distributions.	66
5	External HPC Resources Acquired Throughout Phase 1	81

Acknowledgment

The authors are very grateful to Dr. Jordan Musser and Mr. Justin Weber of NETL, Dr. Jean-François Dietiker of WVURC/NETL and Dr. Tingwen Li of AECOM/NETL for their kind help and valuable discussions.

This research effort is funded by the U.S. Department of Energy’s National Energy Technology Laboratory (NETL) Crosscutting Research Program Transitional Technology Development to Enable Highly Efficient Power Systems with Carbon Management initiative under award DE-FE0026393.

This research used resources of the National Energy Research Scientific Computing Center (NERSC), a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This work has also used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575.

Executive Summary

Two major challenges hinder the effective use and adoption of multiphase computational fluid dynamics tools by the industry. The first is the need for significant computational resources, which is inversely proportional to the accuracy of solutions due to computational intensity of the algorithms. The second barrier is assessing the prediction credibility and confidence in the simulation results. In this project, a multi-tiered approach has been proposed under four broad activities to overcome these challenges while addressing all of the objectives outlined in FOA-0001238 through Phases 1 and 2 of the project. The present report consists of the results for only Phase 1, which was the funded performance period. From the start the project, all of the objectives outlined in FOA were addressed through four major activity tasks in an integrated and balanced fashion to improve adoption of MFIX suite of solvers for industrial use. The first task aimed to improve the performance of MFIX-DEM specifically targeting to acquire the peak performance on Intel Xeon and Xeon Phi based systems, which are expected to be one of the primary high performance computing platforms both affordable and available for the industrial users in the next two to five years. However, due to a number of changes in course of the project, the scope of the performance improvements related task was significantly reduced to avoid duplicate work. Hence, more emphasis was placed on the other three tasks as discussed below.

The second task aimed at physical modeling enhancements through implementation of polydispersity capability and validation of heat transfer models in MFIX. An extended verification and validation (V&V) study was performed for the new polydispersity feature implemented in MFIX-DEM both for granular and coupled gas-solid flows. The features of the polydispersity capability and results for an industrially relevant problem were disseminated through journal papers (one published and one under review at the time of writing of the final technical report). As part of the validation efforts, another industrially relevant problem of interest based on rotary drums was studied for several modes of heat transfer and results were presented in conferences. Third task was aimed towards an important and unique contribution of the project, which was to develop a unified uncertainty quantification framework by integrating MFIX-DEM with a graphical user interface (GUI) driven uncertainty quantification (UQ) engine, i.e., MFIX-GUI and PSUADE. The goal was to enable a user with only modest knowledge of statistics to effectively utilize the UQ framework offered with MFIX-DEM Phi to perform UQ analysis routinely. For Phase 1, a proof-of-concept demonstration of the proposed framework was completed and shared. Direct industry involvement was one of the key virtues of this project, which was performed through forth task. For this purpose, even at the proposal stage, the project team received strong interest in the proposed capabilities from two major corporations, which were further expanded through out Phase 1 and a new collaboration with another major corporation from chemical industry was also initiated. The level of interest received and continued collaboration for the project during Phase 1 clearly shows the relevance and potential impact of the project for the industrial users.

Technical Summary

The Subtopic 4B in DE-FOA-0001238 raised two major challenges that hinder the effective use and adoption of multiphase computational fluid dynamics by industry: (i) the need for adequate computational resources to obtain solutions of sufficient accuracy, and (ii) the need for assessing the prediction credibility and confidence in the simulation results.

The proposed project aimed and was designed to address these two challenges in a cohesive and technically-sound manner through both Phase 1 and then Phase 2 (if awarded) when initially proposed, and offered a consequential step-up in the performance and capabilities of MFI-X-DEM while integrating it with a unified uncertainty quantification framework for seamless and intuitive predictive credibility assessment by industrial users. All seven objectives given in the Subtopic 4B of the FOA (as shown in Figure 1) have been addressed through our project tasks since the start of Phase 1, as we believed that only an integrated and comprehensive action plan can offer a real developmental breakthrough for these non-trivial, interdependent challenges.

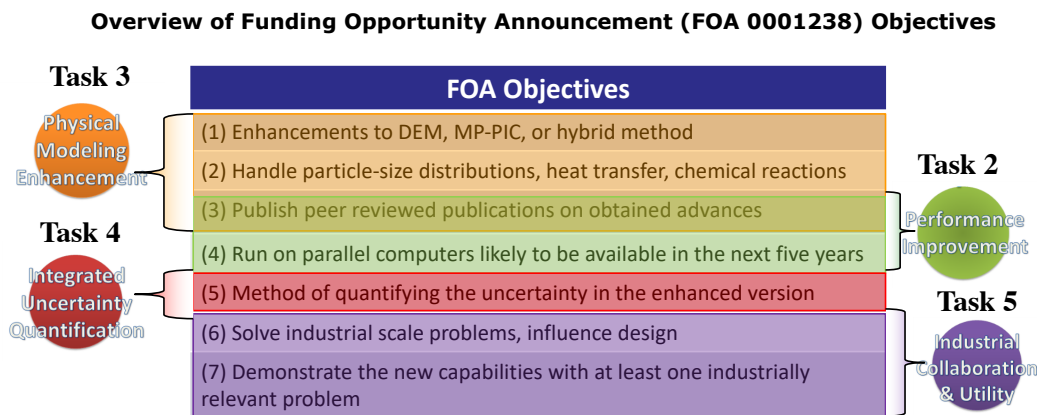


Figure 1. Overview of the Funding Opportunity Announcement (FOA 0001238) objectives and associated subtasks implemented in the project to address these objectives

Figure 2 illustrates a general overview of the planned contributions of the project through performance and accuracy improvements expected for MFI-X, i.e., time-to-solution is reduced substantially while model inaccuracies are reduced through enhanced physical modeling capabilities such as polydispersity implementation or heat transfer model validation. Initially we targeted to achieve between 5X to 20X improvement in performance through various code modernization tasks for effective memory management, good vectorization, and algorithmic

enhancements such as dynamic load balancing, computation/communication overlap, and hardware advances in the targeted HPC systems. However, in course of the project to avoid duplicate work the emphasis on performance improvements were significantly reduced and more emphasis was placed on physical modeling enhancement and remaining other two tasks such as the proof-of-concept demonstration of the integrated UQ framework to enable predictive credibility assessment, which is a critical feature that is currently lacking.

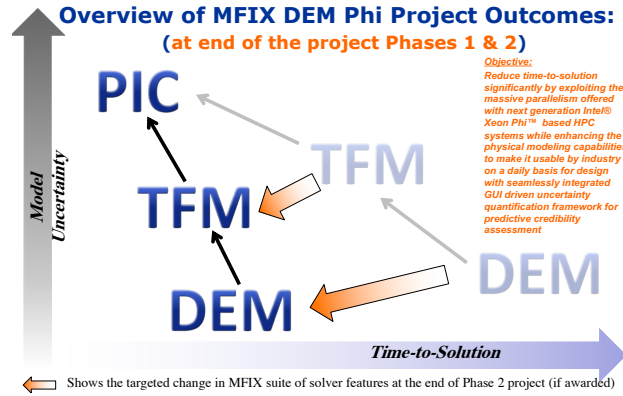


Figure 2. Schematic depiction of the contributions proposed in this project, original illustration adapted from Musser et al. [23]

Figure 3 shows the overall view of the subtasks incorporated at the beginning to achieve the objectives of the project to span both Phase 1, and Phase 2 (if awarded).

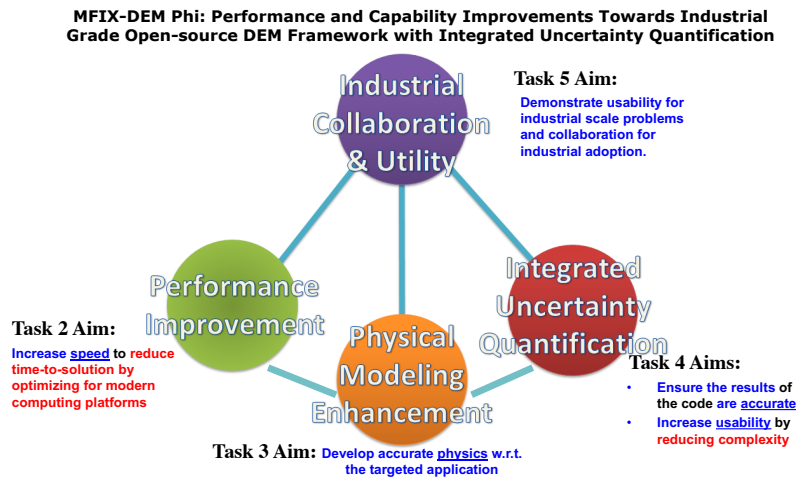


Figure 3. Listing of the Funding Opportunity Announcement (FOA 0001238) objectives and associated subtasks implemented to address these objectives

Although an integrated comprehensive plan was developed aiming the span of both phases, current report presents the contributions and results obtained only for Phase 1 of the project activities, which reflects the originally funded performance period between 09/09/15 to 02/28/17.

Task 1 - Project Management and Planning

The project team included researchers from different disciplines to address the listed FOA objectives. Dr. A. Gel from School of Computing, Informatics, Decisions Systems Engineering (SCIDSE) at Ira A. Fulton Schools of Engineering was the principal investigator for the entire duration of the project and worked in collaboration with co-PIs, Dr. Yang Jiao from Materials Sciences program of School of Matter, Transport and Energy Engineering (SEMTE) and Dr. Heather Emady from Chemical Engineering program of SEMTE. Also the project had two external co-PIs, Dr. Charles Tong from Lawrence Livermore National Laboratory and Dr. Jonathan Hu from Sandia National Laboratories both in Livermore, CA. Hence, various project management and planning processes were implemented by considering the interdisciplinary nature of the participants to improve the communication and effectiveness of the collaboration among the various team members.

Three subteams, one for capability improvements for a general radiation heat transfer framework implementation (Task 2), one for verification, validation and uncertainty quantification for credibility assessment (Task 3), and one for performance improvements (Task 4) were organized. The co-PIs in each subteam held frequent meetings on the specific assigned tasks ranging from weekly meetings at ASU to monthly or quarterly visits/meetings to federal laboratories to work close with the resident co-PI at the site.

The weekly progress update meetings at ASU involved various presentations by the team members and the students to provide education on various technical points and plans for the enhancements to the code. At each meeting, presentation slides were compiled and store to keep track on the status of the relevant milestones, so that progress toward these milestones can be reported accurately in the required project progress reports to the program managers at the U.S. Department of Energy.

In addition to the kick-off and closure presentations, a close contact was maintained with the technical team members at NETL through Webex based progress update meetings initially scheduled bi-weekly and then monthly after first year. Also ASU team attended in-person the annual Multiphase Flow Science Workshop organized by NETL in August 2016 to present the ongoing research activities to Multiphase Flow community and also participate in the poster sessions. The whole project team were enabled access to file storage portal Dropbox and Git repository that was hosted at Bitbucket for effective version control and sharing of research results and developments. In order to improve collaboration and feedback from NETL team members, access to the Git repository was provided to the interested members of NETL's Multiphase Flow Science group for testing the polydispersity implemented version of MFI-X-Dem. This repository is expected to be merged with the main MFI-X GitLab repository at NETL for effective integration of the project outcomes. As part of the ongoing, PI Gel led the efforts to compile competitive proposals for acquiring access to Department of Energy's high performance computing program at NERSC and National Science Foundation's XSEDE program at Texas Advanced Computing Center and San Diego Supercomputing Center. The team also has acquired two Intel Xeon Phi based HPC nodes for local development work, which will be available for these project activities. Hence, the project team members were able to utilize state-of-the-art HPC facilities from the first day without delays for any preparations due to extensive experience at these HPC sites.

Task 2 - Performance Improvements in MFIX-DEM

An important component of the proposed project was performance improvements aiming the high performance computing platforms that will be most likely commonly available in the next 3-5 years for the industrial users. For this purpose, Intel[®] Many Integrated Core (MIC) architecture based HPC systems, in particular Xeon[®] and second generation Xeon Phi[™] (a.k.a. Knights Landing or KNL) were considered for the target computing platforms as we expected these to be the most commonly available and affordable HPC platforms in the near future based on the current trends and major planned deployments at various U.S. Department of Energy and National Science Foundation High Performance Computing centers. Intel MIC was chosen as they also offer a unique feature, i.e., all of the programming can be achieved with familiar languages such as C/Fortran and MPI/OpenMP libraries without the need for another language (e.g., CUDA in the case of NVIDIA[®] GPUs) or libraries (e.g., OpenACC). This feature provided an important advantage for developing and maintaining open-source codes like MFIX-DEM. Performance portability is another benefit that ensures the investments can be extended further.

MFIX offers a unified framework from a single code base for a diverse range of multiphase flows through several modes of operations, such as Two Fluid Method (TFM), Continuum Discrete Method (CDM) and Discrete Element Method (DEM) [5],[13].

As previously discussed, Figure 2 illustrates a general overview of the originally planned contributions of the project through performance and accuracy improvements expected for the code, i.e., time-to-solution is reduced substantially while model inaccuracies are reduced through enhanced physical modeling capabilities. For this purpose, the project aimed to address performance issues that arise in CDM (a.k.a. coupled CFD and DEM) and DEM modes for a balanced approach. Hence, the subtask was divided into performance improvement for two major solution processes offered by MFIX-DEM: (i) Dispersed Phase (Lagrangian) Solution Process, i.e., the DEM Solver, and (ii) Continuum Phase (Eulerian) Solution Process, i.e., the CFD Solver. As the problem size increases towards industrial scale, both (i) and (ii) affect time-to-solution in CDM mode, whereas the pure DEM mode is affected by (i) only.

In order to achieve the reduction in time-to-solution as illustrated in the orange colored shift shown in Figure 2, there is a need to address both performance bottlenecks in a balanced approach. The proposed improvements have been split into two separate subtasks, which were initiated in Phase 1 and proposed to be carried out extensively in Phase 2. However, during the course of the project the work scope for this subtask was significantly reduced with the approval of DOE/NETL to avoid duplicate effort with another institution who was awarded to work on performance improvement topics for a longer duration. Also DOE/NETL decided to terminate a particular subaward/subtask on performance improvements in December 2017.

Hence, as part of the updated project management plan which reflected the significantly reduced scope in Task 2, we first present some of the preliminary performance profiling results prior to this change to document the observed bottlenecks in single core code performance at that time without the polydispersity implementation. Then the performance profiling results for runs conducted on single core using the same tool (i.e., Intel Advisor XE) are

presented for the new version of the MFI-X-DEM to better illustrate whether polydispersity implementation, which was developed as part of this project towards physical modeling enhancements subtask has introduced any new bottlenecks in DEM solution process.

Prior Performance Profiling Results Without Polydispersity

Several representative benchmark problems were identified in consultation with our industrial collaborators and also NETL team members at the beginning of the project to conduct performance profiling and use the results to establish the baseline performance for documenting improvements. In order to achieve consistent performance profiling, the benchmark problems were prepared to reach a certain time level in simulation and profilings were always conducted from the same initial restart conditions generated as a result of these preparation activities. For example, the rectangular 3D riser problem was simulated till 5 seconds to let the transients pass and then always restarted from 5.0 seconds to run for 0.1 seconds.

The performance profiling was performed with a recent tool developed by Intel[®], i.e., Advisor (also known as "Advisor XE", "Vectorization Advisor" or "Threading Advisor") [10]. This performance profiling tool appears to aim the developers for code modernization by deeply analyzing application software to find answers to the questions like "Are the loops that display bottleneck performance vectorized? If not then what is the constraint? If yes then is it using the optimal instruction set targeted for the architecture (e.g. AVX512 for Intel Knights Landing processors)?" Although such information could also be obtained through other performance profiling tools that was employed in the project (i.e., Intel[®] VTune[™] Amplifier XE [11] or TAU from University of Oregon [15, 2]), what made Advisor preferable over other tools was the customized recommendations generated at the end of the analysis to improve the vectorization capability of the code for the targeted hardware architecture. Considering the substantial SIMD instruction capability offered with modern Intel Xeon and Xeon Phi based architectures, achieving peak vectorization performance is one of the essential and sometimes lowest hanging fruit for performance improvements. Hence, due to our collaboration with Intel engineers and lead developer of Advisor, opportunity to become one of the beta testers for the new tool to test the features offered at the early stage of the project offered a unique benefit as our project specifically targeted Intel[®] Xeon and Xeon Phi architectures.

Although the scope of the current subtask was significantly reduced with DOE/NETL approval due to the changes mentioned earlier, some of the preliminary results from Intel Advisor profilings acquired prior to this change is presented in this section for illustration purposes on what level of information can be acquired from these types of tools and how they can improve the productivity in code modernization. Figures 4 to 9 show the Intel Advisor profiling results for the Benchmark Case # 1, which is the 3D rectangular riser where initially TFM solver dominates but then DEM solver is predominant and demonstrates some bottlenecks. Figure 4 shows that non-vectorized scalar code dominates 66% of the overall time spent by the loops during the execution of the code for this problem and only 34% of the time spent was in the vectorized loops. Even the latter loops, which were vectorized

to take advantage of the SIMD hardware, the vectorization efficiency wasn't good. This situation most likely indicates the proper vectorization instructions (e.g., AVX2 for Xeon or AVX512 for Xeon Phi processors) are not being utilized. Figure 5 is the continuation of the previous figure where at the bottom the top time-consuming loops are specifically outlined, and recommendations for performance improvement by targeting specific loops are displayed. For example, as expected `calc_force_dem` subroutine was determined to be the subroutine having at least one loop that consumes most of the time among all other loops during the execution in spite of the low trip count. When compared to the VTune Amplifier XE profiling results (as presented in 2016 Q3 quarterly report), same top two subroutines were listed as the primary bottleneck but VTune required further analysis to understand the root cause whereas Advisor was able to point the potential root causes at the loop level if they are due to poor vectorization.

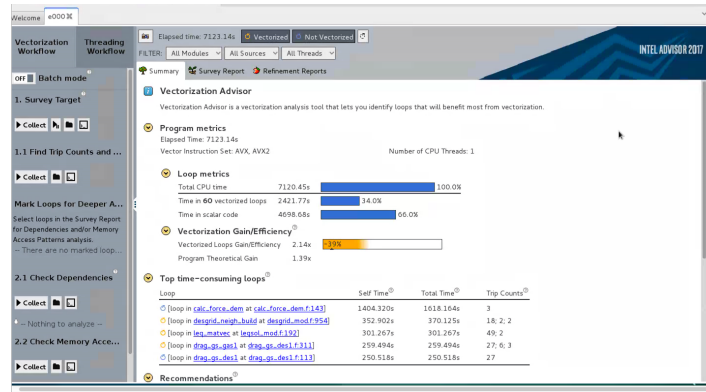


Figure 4. Dashboard screen showing the performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.

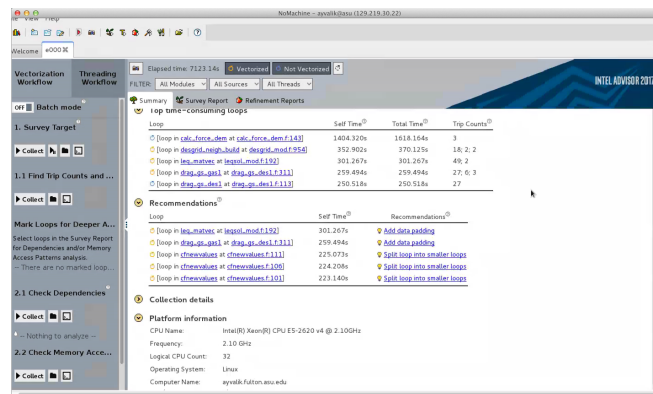


Figure 5. Intel Advisor tool based performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) showing the top loops that consume most of the time during the execution session.

Figure 6 shows the location of the loop listed as top-time consuming with the source code and some measured metrics. For example, cycling the loop due to the condition being satisfied has taken substantial portion of the execution of this loop due to scalar execution. Most likely the conditional branching has caused the compiler not to vectorize the loop and force scalar execution. This situation can be verified by reviewing the compiler optimization report, which could be easily generated with the compiler level flags such as `-qopt-report=5` during compilation.

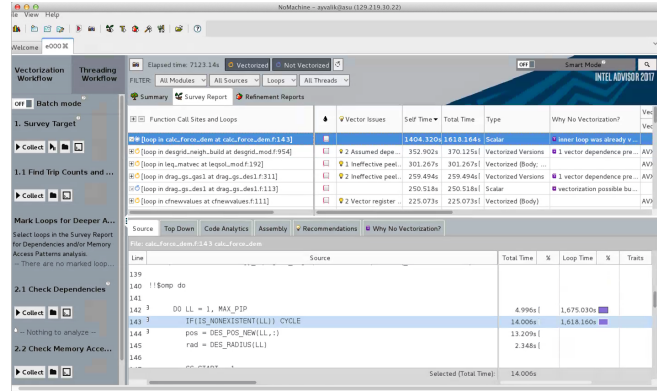


Figure 6. Intel Advisor screen displaying the location of the top bottleneck loop in the source code with some execution metrics such as loop time.

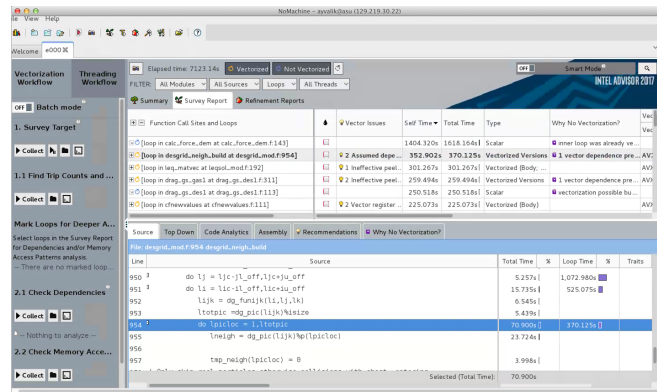


Figure 7. Display of the second top time-consuming loop during the execution of Benchmark Case 1 on Intel Xeon Haswell (HSW)

Figure 7 shows the second top time-consuming loop, which appears to be in DES-GRID_NEIGH_BUILD, and same subroutine was also listed in VTune Amplifier analysis earlier. In the Advisor screen the explanation for the poor performance in this particular loop is associated with vector dependency, which is most likely due to a variable defined with in the

loop and relying on another array triggering the dependency. Although loop appears to be vectorized up to a certain extent the assessed dependency causes a scalar loop instruction to be generated as it is shown in the breakdown list in Figure 8

For more advanced users, Advisor also offers the option to analyze the hot-spot location (i.e., the loop identified as bottleneck during execution) in compiler generated assembly languages as shown in Figure 8. This capability is very useful for those who are familiar with the instruction set of the targeted architecture and can immediately diagnose the problem through to the instructions generated by the compiler. More detailed text based assessment of the loop vectorization and associated suggestions for improvement can be obtained in text mode as shown in Figure 9, which clearly shows scalar loop causing the inefficiency with the rest of the vectorization. One of the solutions is to remove the dependency by explicitly declaring a variable locally and setting to the arrays at beginning of the subroutine so the compiler can determine the values of the array to be known while the loop is executing.

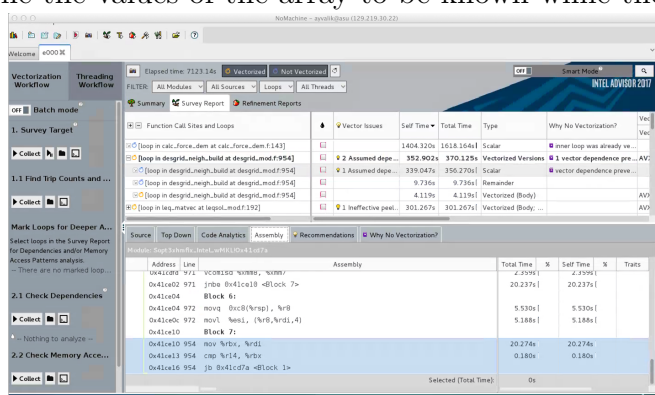


Figure 8. Compiler generated assembly language view of the second top time-consuming loop determined by Intel Advisor.

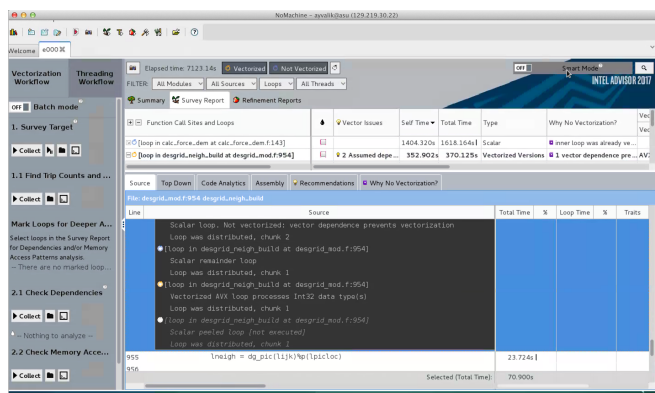


Figure 9. Performance profiling of Benchmark Case 1 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.

Benchmark Case # 2 Hopper Discharge: The second representative benchmark case selected was the hopper discharge, where we have created an initial configuration of hopper filled with granular material, which was settled to make sure most of the kinetic energy is dissipated. For the performance profiling the bottom of the hopper was opened to let the discharge begin and profiling was performed for certain duration to capture the typical behavior of granular flow. For the initial profiling, we performed the simulation till the hopper is discharged to establish a baseline and compare against shorter wall-clock runs to make sure the hot-spot characteristics observed do not change based on the duration of the simulation. In contrast to the first benchmark case, this problem only utilized the DEM solver in MFX as only granular material flow is simulated by employing the Lagrangian framework based discrete element method solution. First performance profiling was performed with Intel VTune on single core to get an overview of the bottlenecks at the processor level. Hence, Figure 10 shows the preliminary performance profiling results. This time `DESGRID_NEIGH_BUILD` appears to be the top time-consuming subroutine whereas `CALC_FORCE_DEM` is the second ranked.

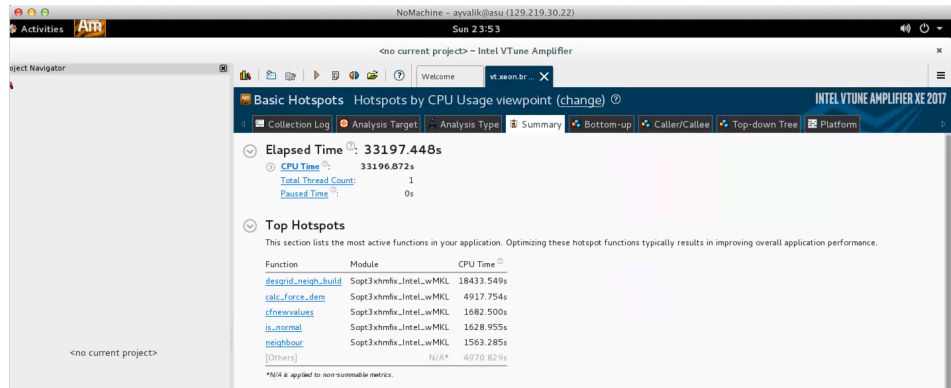


Figure 10. Summary display for performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the Intel VTune tool to analyze general performance bottlenecks.

Figures 11 to 12 show the detailed view provided by VTune Amplifier on the location of the top time-consuming bottleneck in `DESGRID_NEIGH_BUILD`, which appears to coincide with the same loop as highlighted in Case #1 with Intel Advisor reports (as shown in Fig. 7).

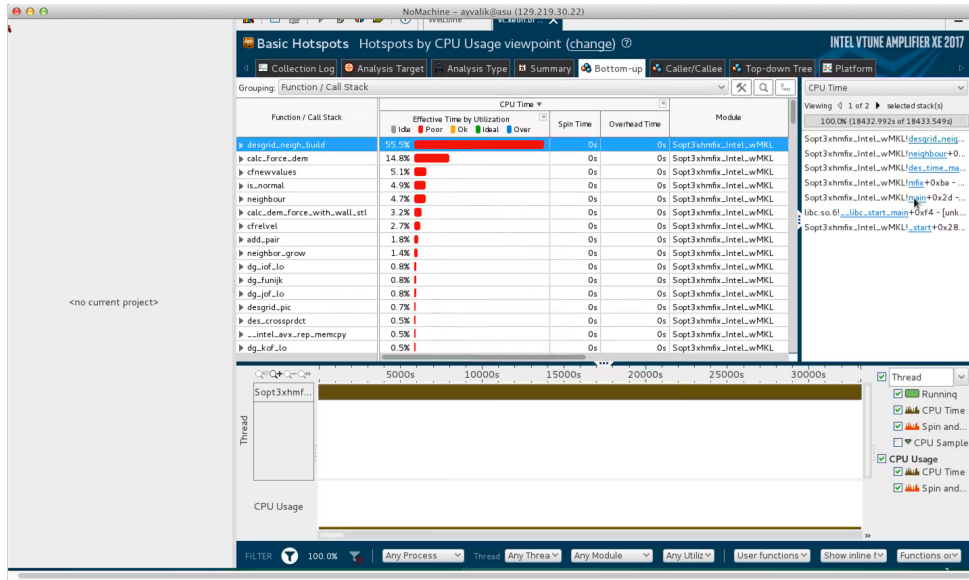


Figure 11. Performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the Intel VTune Amplifier tool showing the most time consuming subroutines.

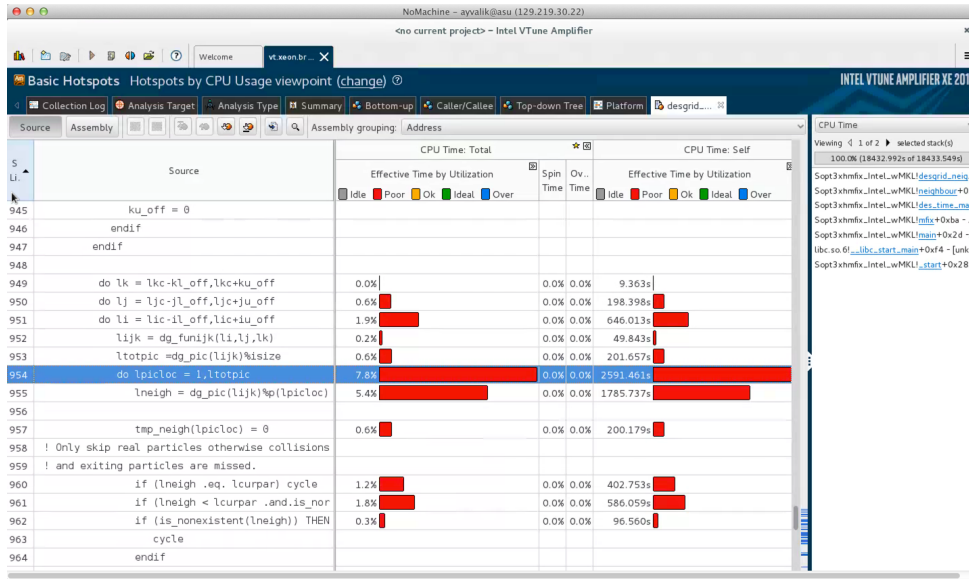


Figure 12. Detailed view of the hot-spots in the top time-consuming subroutine DESGRID_NEIGH_BUILD for Benchmark Case 2 as identified by Intel VTune Amplifier.

Same set of simulations were profiled with Intel Advisor also to better isolate the performance hot-spots that are directly associated with poor vectorization. Figures 13 and 14

show the Advisor reports and findings for the same benchmark case. As opposed to the Case #1 Advisor report (shown in Fig. 4), Case # 2 appears to spend nearly half of its time in vectorized loops but still half of the code is running in scalar loops. This situation inevitably implies leaving substantial performance on the table as we are not taking advantage of what the hardware has to offer if properly vectorized.

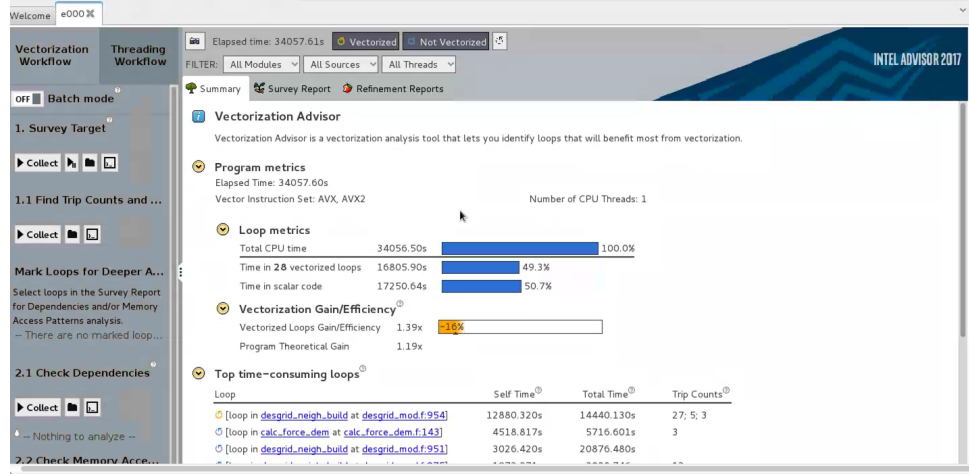


Figure 13. Summary report for performance profiling of Benchmark Case 2 on Intel Xeon Haswell (HSW) with the new Intel Advisor tool to identify the vectorization related bottlenecks.

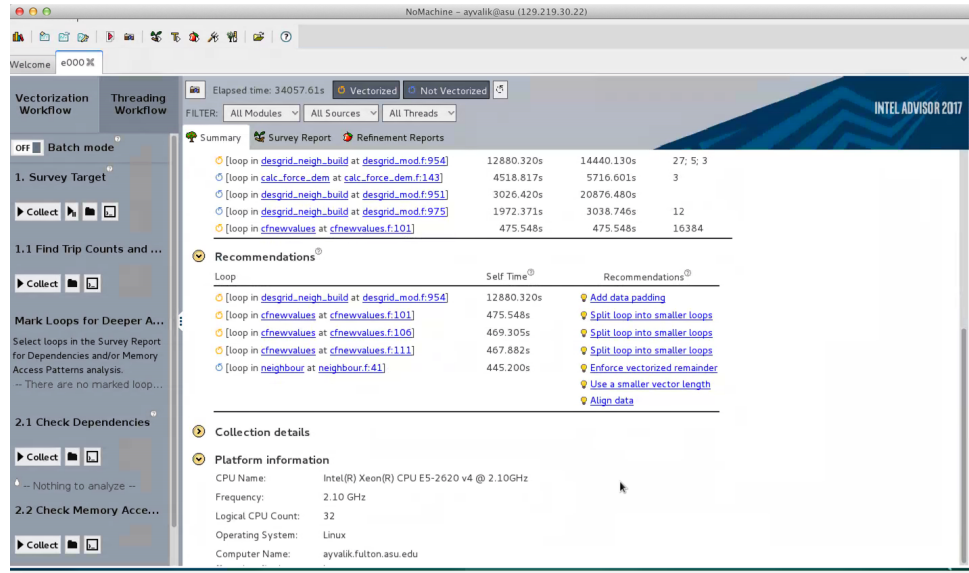


Figure 14. Detailed view of the top time-consuming loops determined during performance profiling of Benchmark Case 2 with Intel Advisor.

Most Recent Performance Profiling Results With Polydispersity

Figures 16 and 17 show the performance profiling results for one of the industrially relevant problem, i.e., benchmark case # 2, which was initially proposed by our industrial collaborator, Procter & Gamble and then extended for the particle height classification problem proposed by ExxonMobil collaborator. The revised benchmark problem is a hopper discharge configuration with binary spherical region as shown in Figure 15. Two types of particles are considered with a density ratio 3.0 between spherical region and the rest of the hopper. Total particle count was 200,000 particles where approximately 50,000 were in the spherical region with a uniform distribution of particle size with D in $[0.05, 0.15]$ cm for both the spherical region and the rest.

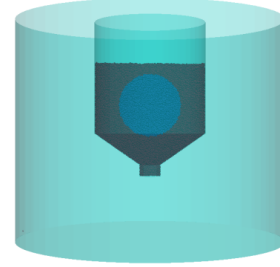


Figure 15. Revised benchmark problem

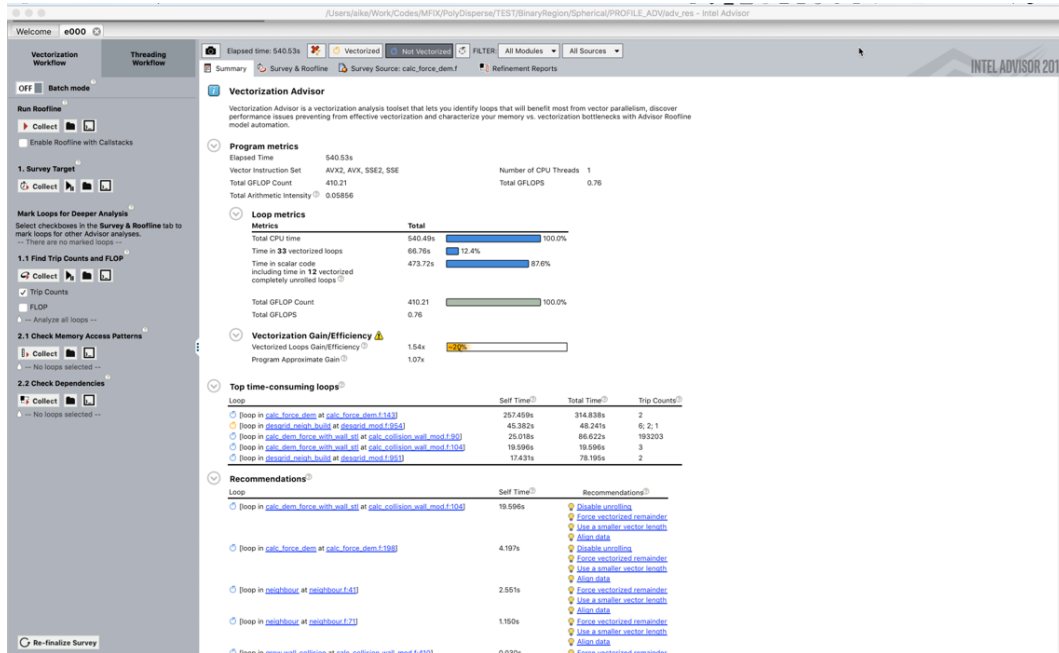


Figure 16. Intel Advisor Profiling of Polydispersity implemented 2016-1 version of MFI-X-DEM

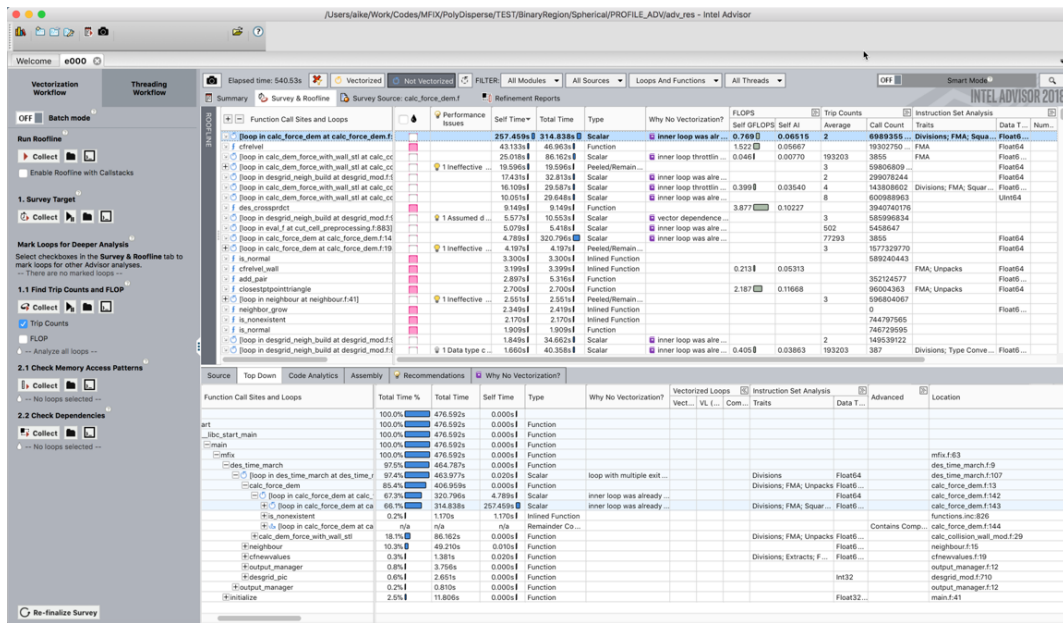


Figure 17. Detail view of Intel Advisor Profiling of Polydispersity implemented 2016-1 version of MFI-X-DEM

As seen in Figure 16, most of the computational time is in `CALC_FORCE_DEM` and `DES-GRID_NEIGH_BUILD` routines, which were the same set of routines without polydispersity feature. However, this time `CALC_FORCE_DEM` appears to be top subroutine, which is possibly due to the fact the duration of execution for data collection were different for both benchmarks as it can be seen from Self Time column in Figures 14 and 17. No additional performance profiling results or implementation of the suggested improvements are presented due to the DOE/NETL approved change in work scope for this subtask.

Task 3 - Physical Modeling Capability Enhancements in MFIX-DEM

In order to address the FOA objectives addressed earlier in Figure 1, a standalone task was adopted to enhance the physical modeling capabilities of the computational models for more accurate solution of the real engineering problems encountered by the industry. For Phase 1, the capability enhancements task were divided into two major subtasks. The first subtask aimed to improve the capability MFIX-DEM in handling particle size polydispersity to allow multiple solid phases, each possessing a distinct arbitrary particle-size distribution for efficient and realistic multiphase flow simulations. The second subtask was aimed to validate the existing heat transfer models in MFIX-DEM since these models have not been previously tested with more than two particles and a published study existed at the beginning of the project as September 2015. The following sections present the results of these subtasks.

Subtask 3.1 - Enhance the Capability for Handling Particle Size Distributions

The objective of this subtask is to significantly improve MFIX-DEM's physical modeling capability for handling particle size polydispersity. This objective has been successfully achieved by implementing new data structures and sub-routines in MFIX source codes. The new features have been extensively verified and validated using hopper bin discharge studies [8]. The enhanced MFIX-DEM codes with the new polydispersity-handling features have been utilized in a particle height classification problem in collaboration with our industrial partner at ExxonMobil [7].

3.1.1 - Implementation of Polydispersity Features in MFIX-DEM

Our implementation for polydispersity aimed to minimize the invasive structural changes in the algorithm and the original MFIX-DEM code. In the latest release of MFIX-DEM (i.e., 2016-1), each distinct solid phase is only allowed to possess a unique particle size, which is saved in the array `D_P0`, whose size is the maximum number of solid phases in the system. Each solid particle is assigned a unique phase index based on its diameter, which helps in phase-specific physical properties of the particles (e.g., material density, Young's modulus, coefficient of friction, coefficient of restitution, etc.) to be retrieved and employed in the subsequent computation of particle contact forces.

The basic idea of our new implementation is to separately save the particle geometrical parameters (i.e., diameters), which are particle-specific and the phase-specific physical parameters. In particular, we modified the array `D_P0` such that it saves the diameter of each particle in the system, thus possessing a size of the total particle number N . For example, `D_P0(i)` now gives the diameter of particle i , instead of the diameter of particles for phase i . Accordingly, the phase index of each particle is not assigned according to its diameter, but assigned based on its physical properties, e.g., material density. The change of data

structure for saving and retrieving individual particle diameter (or radius) affects a number of subroutines which utilize `D.P0(i)` as input for subsequent calculations. The modified subroutines include those for particle force/torque calculation (`calc_force_dem.f`), solid-gas momentum transfer calculation (`calc_drag_des.f`), grid cell locator (`desgrid_mod.f`) and neighbor locator (`neighbour.f`). We note the modifications do not affect any of the underlying governing equations for the system nor specific contact force models to be utilized. Also as shown as part of the Task 2 results, the modifications haven't adversely affected the performance characteristics of MFIX-DEM by creating new performance bottlenecks.

The new implementation offers three types of built-in probability distribution functions (PDF) for size, including normal, log-normal and uniform distributions. However, the addition of new distributions can be easily done through the `DES/RANDOMNO_mode.F` subroutine. For a user-specified distribution, the particle diameter information is provided in the restart files (*.RES) and is read in during the initialization stage. In addition, users can specify the particle size distribution for both the initial condition (IC) and mass-in-flow (MIF) boundary conditions (BC). This allows the simulation of complex multiphase flow processes during which additional solids are injected into the system. This implementation is achieved by introducing two sets of new keywords, respectively, for ICs and MIF BCs. The new polydispersity feature is expected to be available in the next release version of MFIX with detailed documentation and tutorial cases.

When utilizing the polydispersity features, it is recommended that the ratio between the largest and smallest particle diameter is smaller than 5, i.e., $D_{max}/D_{min} \leq 5$. First, the computational cell grid size L is determined by D_{max} via $L = \lambda D_{max}$ (where $\lambda = 1.1$ by default). A very large value of particle size ratio would lead to a large computational cell with too many smaller particles, which in turn decreases the computational efficiency. In addition, for larger size ratios, the effects of small particles on a large particles can be very well described by the continuum approach and by treating the small particles as a continuous fluid phase.

3.1.2 - Verification and Validation (V&V) Using Hopper Bin Discharge Studies

(a) Verification of the polydispersity implementation:

As part of our preliminary V&V effort, we first verified the new implementation by simulating particle discharge from a 3D hopper with equal-sized spherical beads. The 2016-1 release version of MFIX is also used to simulate the same system for comparison. The hopper contains a cylinder part of height 5 cm and diameter 6 cm, connected to a cone part with height 5 cm and angle of 30 degrees. The bottom of the hopper cone is further connected to a short cylinder tube of height 0.2 cm and diameter 0.5 cm. The monodisperse beads possess a diameter of 0.068 cm, spring constant of $2.5 \times 10^5 g/s^2$, friction coefficient of 0.01 and coefficient of restitution of 0.5.

The initial configuration for the discharge simulation is obtained by first filling the hopper with 15540 beads from the top (through the mass-in-flow boundary condition), while the

bottom is kept closed (through the no-flow boundary condition). The resulting packing is allowed to settle completely, to dissipate the kinetic energy. The stable packing at the end of settling is used as the initial configuration for the discharge simulation until the hopper is fully discharged.

Table 1. Verification simulation details for the discharge hopper with monodisperse spherical beads.

	Domain Decomposition Configuration	Total Number of Particles	CPU hours
MFIX-DEM 2016-1	$2 \times 2 \times 2$	15544	5.45
Our implementation	$2 \times 2 \times 2$	15540	5.44

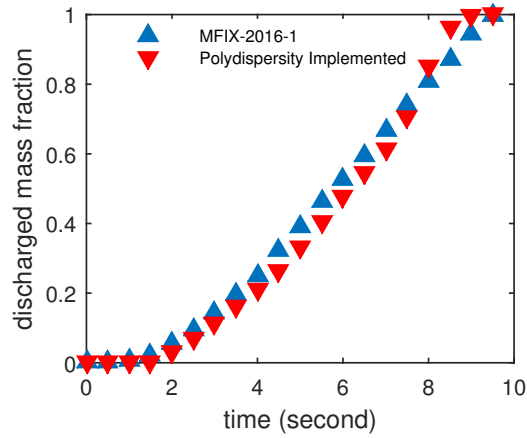


Figure 18. Discharge dynamics for a 3D hopper with equal-sized spherical beads. The discharged mass vs. discharge time curves obtained from both the 2016-1 MFIX and our new implementation agree well with one another.

Tab. 1 summarizes the simulation details, including the number of particles, domain decomposition configuration and total CPU hours required for full discharge. Fig. 18 shows the discharged mass vs. time for the monodisperse system, obtained using the 2016-1 MFIX-DEM release and our new polydisperse implementation derived from the 2016-1 release. It can be seen that in both cases, the initial discharge rates are relatively small and stabilize within ~ 2 seconds after discharge starts. The discharged mass fractions for the two cases slightly deviate from one another due to uncertainty in the initial packing configurations. The overall discharge dynamics agree very well with one another. Also, the total computational cost in terms of wall-clock CPU hours for the simulations are also comparable in the two cases. Hence, these results verify the correctness of our new implementation.

(b) Hopper Discharge Experiments:

In the next step of our V&V effort, to validate our polydispersity implementation hopper discharge experiments were carried out using two types of polydisperse particles with varying initial packing configurations. Detailed discharge dynamics were obtained for both types of particles for comparison with simulation results. The hopper used in the experiments (shown in Fig. 19) is 3D printed using a Stratasys Dimension 1200es SST (Stratasys Ltd. MN, USA), with ABSplus thermoplastic. It contains a cylinder of 12.5 cm height and 12.5 cm diameter, connected to a cone with a height of 3.5 cm and 55° cone angle. The bottom of the hopper cone is further connected to a short cylinder of height 1.3 cm and diameter 2.5 cm. The hopper is leveled and clamped to a support stand. A glass slide gate is used to close and open the hopper outlet, to fill and discharge, respectively.



Figure 19. Experimental setup of hopper: Clamped 3D printed hopper with a beaker, for collecting discharged particles, placed on an analytical balance to measure the mass of discharged particles for sampling.

The granular materials used in the experiments are silica beads (soda-lime silica glass). The silica beads are purchased from Potters Industries, PA, USA; with particle density 2.5 g/cm^3 in different sizes. The particles are further sieved to narrow the particle size distribution (PSD). The sieved particles are analyzed using a Malvern Morphologi G3SE (Malvern Instruments Ltd, UK), and two distinct sized polydisperse silica beads with a bimodal size distribution (shown in Fig. 20) are used for the experiments. Specifically, the system can be considered to contain two phases of solid particles, each possessing a distinct normal size distribution, with the same physical properties. The fine particles possess a normal distribution with a mean of 1.5 mm and standard deviation of 0.3 mm, and the coarse particles possess a normal distribution with a mean of 2.9 mm and standard deviation of 0.1 mm (see Fig. 21).

While the particles discharge through the hopper, they become tribocharged and tend to stick to the walls of the hopper. To remove any charging of the particles during discharge, the particles are treated with an anti-static solution. The anti-static solution is prepared by dissolving 1 ml of ASA antistatic agent (Electrolube, UK) into 100 ml of ethanol. No change in the discharge dynamics and segregation is observed before and after the particles are coated with the anti-static solution.

The masses of the fine and coarse particles used in the experiments are 580 g and 420 g, respectively. Two initial packing configurations are prepared using the fine and coarse particles. In the first configuration, the fine and coarse particles are well mixed. For this, both the fine and coarse particles are divided into 10 equal portions. One portion of fine particles is mixed with one portion of coarse particles using a Turbula T2F Shaker-Mixer (Glen Mills Inc., NJ, USA). The same procedure is repeated with the remaining 9 portions to produce 10 equal portions of well mixed fine and coarse particles. The hopper outlet is closed, and each portion is loaded slowly into the hopper, to minimize segregation due to free fall. The particle bed is leveled after each portion is loaded. In the second configuration, the coarse particles are first packed at the bottom of the hopper (with the outlet closed). Then the fine particles are loaded on top of the coarse particles to form a layered packing. Each layer is leveled after loading into the hopper. We note that these configurations were also used by Ketterhagen et al., (2007) [21].

The discharge experiment is carried out using a discontinuous sampling method. The outlet of the hopper is opened, and samples of equal mass are collected using the discontinuous sampling method. The initial and final arrangement of particles, for both mixed and layered configurations, are shown in Fig. 22. The collected samples are sieved and weighed to determine the mass of fine and coarse particles. Then the discharged mass fraction of both the fine and the coarse particles are determined, and the segregation data is plotted as the normalized mass fraction of fines (γ_f^N) and the overall discharged fractional mass ($\bar{\gamma}$), using Equations [2.1 and 2.2].

$$\gamma_f^N = \gamma_f^{discharge} / \gamma_f^0 \quad (2.1)$$

$$\bar{\gamma} = m^{discharge} / m^0 \quad (2.2)$$

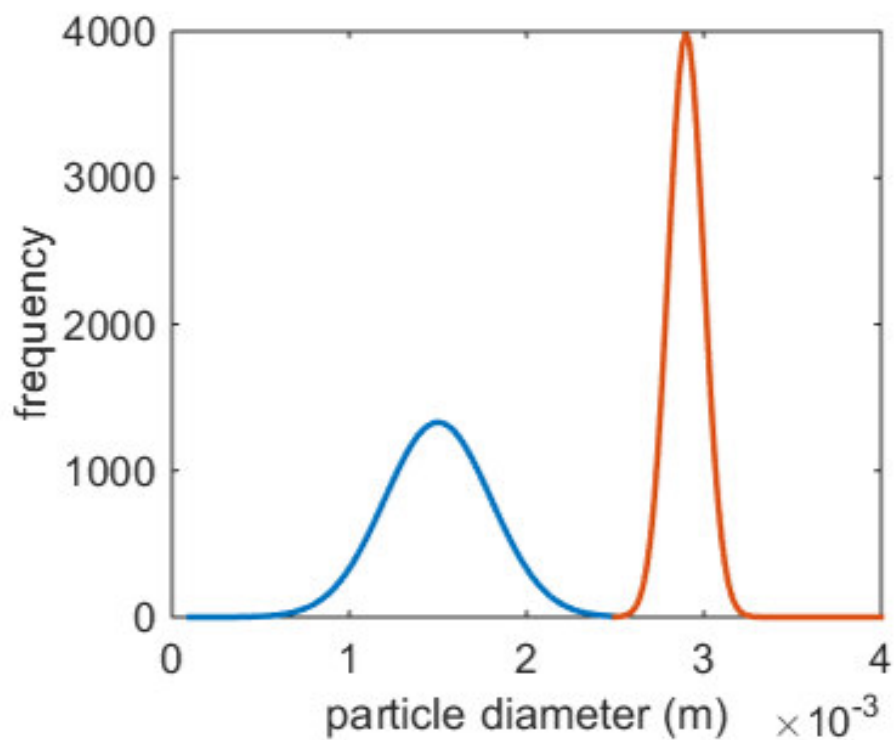


Figure 20. A bi-modal particle size distribution (i.e., number frequency vs. particle sizes) was employed, which includes two normal distributions for fine and coarse silica bead particles.

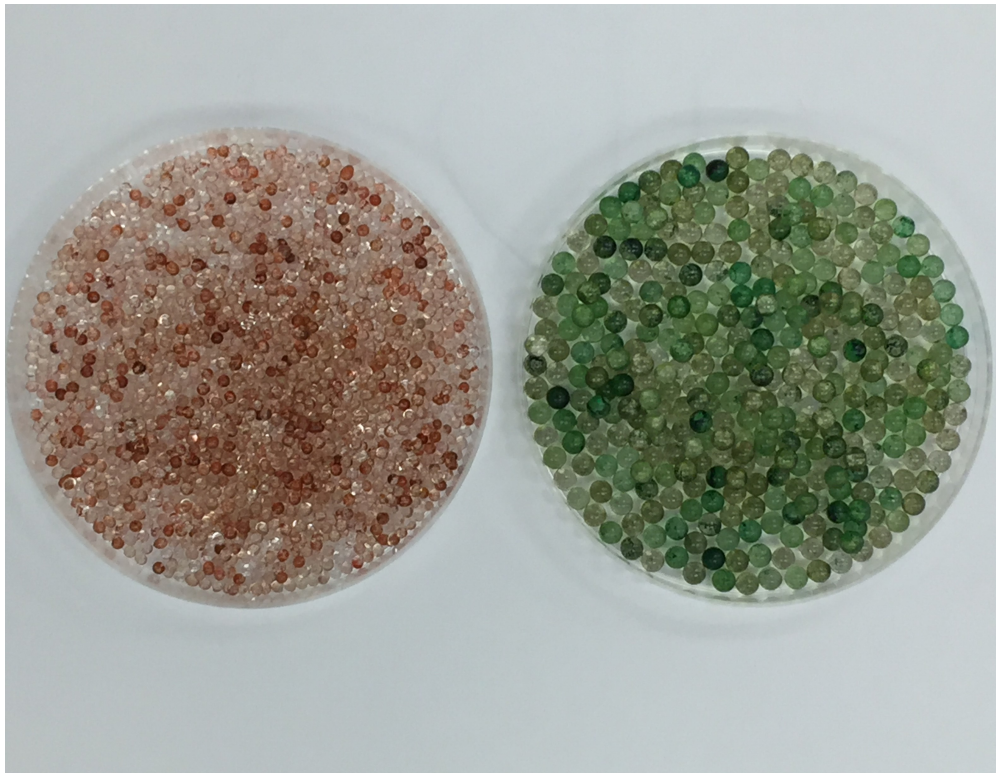
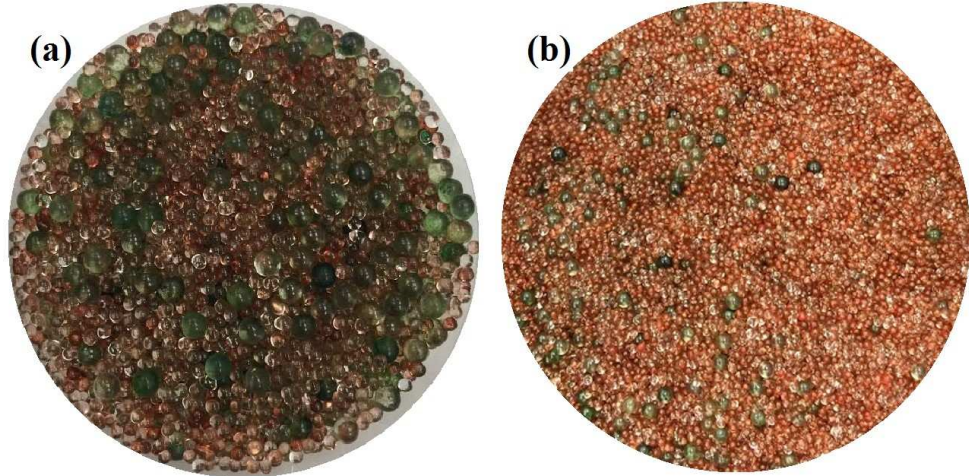


Figure 21. Silica beads: Normal distribution of the fine particles with mean of 1.5 mm and standard deviation of 0.3 mm, colored in red; and the coarse particles with a mean of 2.9 mm and standard deviation of 0.1 mm, colored in green.

Well-Mixed Configuration



Layered Configuration

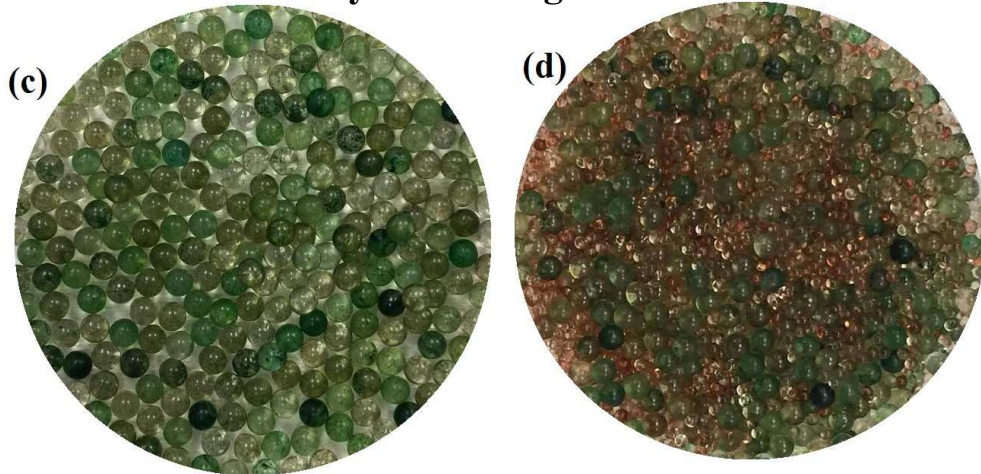


Figure 22. Particle arrangement: (a) First discharged sample from the well-mixed configuration. An approximately equal number of fine and coarse particles are discharged. (b) Last discharged sample from the well-mixed configuration. A majority of fine particles are discharged. (c) First discharged sample from the layered configuration. Only coarse particles are discharged. (d) Last discharged sample from the layered configuration. A mixture of both fine and coarse particles are discharged.

where, γ_f^N is the normalized fine mass fraction of the discharged particles, $\gamma_f^{discharge}$ is the fine mass fraction of the discharged particles, γ_f^0 is the fine mass fraction in the initial packing configuration, $\bar{\gamma}$ is the overall discharged mass fraction, $m^{discharge}$ is the total discharged mass and m^0 is the mass of the initial particle packing. During the experiments with the well-mixed configuration, segregation occurs while loading. Therefore, each experiment is replicated five times to minimize the effects of any non-homogeneous regions on discharge segregation results, for both well mixed and layered configurations. The experimental results are reported in the subsequent section to validate our polydispersity implementation.

(c) Comparison of Experiment and Simulation Results:

In this section, we present the validation of our newly implemented polydispersity feature by replicating the experimental setup in discussed later under Sec. (b). For the simulations, we consider that the glass beads possess model properties consistent with the previous studies [21]. As discussed earlier, in order to handle size distribution more efficiently in the simulation, a maximum value and a minimum value are specified for each size distribution. The fine-to-coarse particle mass ratio is 0.58:0.42, corresponding to a fine-to-coarse number ratio of 8: 1. The parameters for the particle phases used in our simulations are summarized in Tab. 2. We note that for both solid phases, we specify an upper and lower bound for the particle size PDF, to avoid generating non-realistic large/small particles. In our subsequent simulations, the initial packings are generated using the mass-in-flow (MIF) boundary condition (i.e., the particles are filled into the hopper from the top). In the current implementation, the MIF boundary condition does not allow one to precisely control the number of particles for each solid phase to be filled in the hopper. Therefore, the mass ratio is first determined in the hopper discharge simulation and then reproduced in the experiments, in which the mass for each phase can be controlled more precisely. The simulations are carried out on Stampede at the Texas Advanced Computing Center (TACC), which consists of 64-bit Xeon E5-2680 2.7GHz (turbo, 3.5) cores. The code is compiled with mvapich2/2.1.

Table 2. Geometrical parameters, masses of the two solid particle phases and global parameters used in validation simulations.

Sample	Fine	Coarse
Mean Dia.	0.15 cm	0.29 cm
Max Dia.	0.17 cm	0.31 cm
Min Dia.	0.13 cm	0.27 cm
Std. Dev.	0.03 cm	0.01 cm
Total Mass	580 g	420 g
Spring Constant	$2.5 \times 10^5 g/s^2$	$2.5 \times 10^5 g/s^2$
Friction Coefficient	0.5	0.5
Coefficient of Restitution	0.9	0.9

(d) Discharge hopper with polydisperse particles in well-mixed configuration:

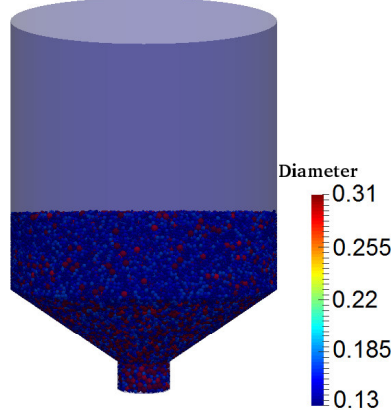


Figure 23. Initial configuration of the discharge hopper containing a well-mixed packing of spherical beads with a bi-modal size distribution. The unit of the particle diameter is centimeter.

We first consider the well-mixed configuration (see Fig. 23). The initial configuration for the discharge simulation is obtained by filling the hopper with both fine and coarse beads together from the top through the mass-in-flow boundary condition, while keeping the bottom closed with a no-flow boundary condition. The injection velocities of the fine and coarse beads through the top are 16 cm/s, with a volume fraction of 0.0145 for coarse beads and 0.0165 for fine beads. The resulting packing is allowed to settle to completely dissipate the kinetic energy, and the resulting well-mixed stable packing is used as the initial configuration for the discharge simulation until the hopper is fully discharged. The number of computational grids used in this simulation is 30 by 30 by 40 respectively along the two lateral directions and the axial direction of the hopper. We note although the grid resolution needs to be specified for MFIX logistics, the actual fidelity is determined by the number of particles employed for the simulation when running MFIX-DEM in granular-flow-only mode. The domain decomposition configuration is 4 by 4 by 4 with 64 cores, and the total CPU hours for the discharge simulation are 6.73 hours.

Fig. 24 shows the snapshots of the discharge simulations at several different times. The discharge dynamics are quantified using the normalized fine mass fraction, γ_f^N , of the discharged particles vs. the overall discharged mass fraction, $\bar{\gamma}$, which are calculated by using Equations [2.1 and 2.2].

In the ideal case that during the entire discharge process, the packing configuration stays well mixed, the resulting $\gamma_f^N - \bar{\gamma}$ curve would be a simple constant function with value 1. However, due to segregation effects, the fine particles cluster at the hopper bottom towards the end of the discharge, leading to an increase of γ_f^N towards the end of discharge (see Fig. 25). As can be seen from the figure, our simulated $\gamma_f^N - \bar{\gamma}$ curve agrees very well quantitatively with our experimental data.

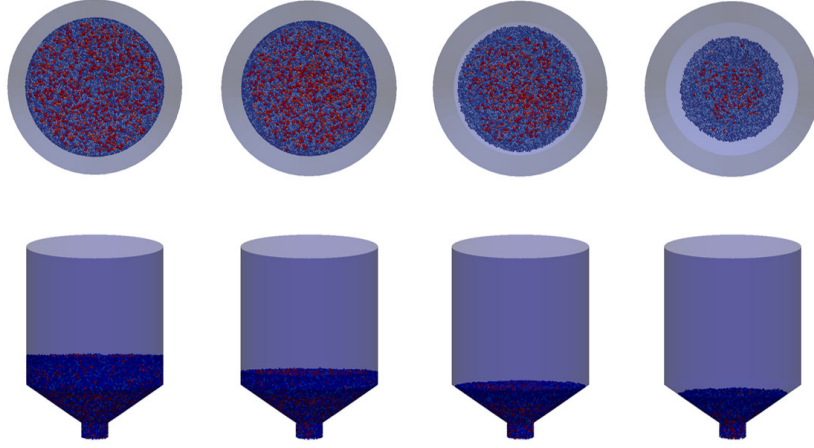


Figure 24. Snapshots of the discharge simulation of the well-mixed configuration at 6 s, 9 s, 11 s and 12 s, respectively, from left to right. The upper panels show the top view and the lower panels show the side view.

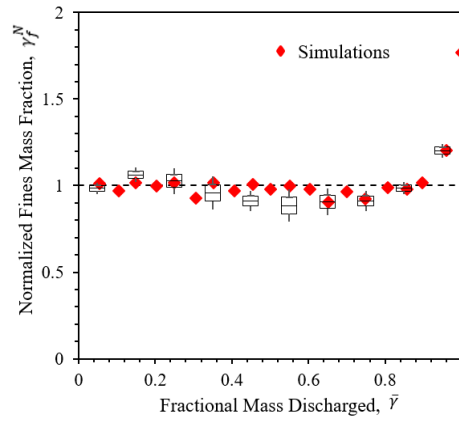


Figure 25. Discharge dynamics are quantified using the normalized fine mass fraction γ_f^N of the discharged particles vs. the overall discharged mass fraction $\bar{\gamma}$ for the well-mixed configuration. The experimental results are represented as a box plot.

Nonetheless, it can be seen that the experimental data tends to deviate more from the ideally mixed case (i.e., the constant function), indicating a stronger segregation effect during the discharge. We note that in an “ideally mixed case”, there will be no aggregation of particles of similar sizes during the entire discharge process. This situation is not possible in practice, as the particles during a discharge process always gain kinetic energy and thus, inevitably form aggregations of different sizes. The observed discrepancy is possibly because of the existence of locally segregated regions in the experimentally prepared initial packing configuration due to particle size dispersity, which is inevitable even though very careful mixing was conducted in the experiment. On the other hand, the particle mixing in the initial configuration can be much better controlled in the simulation through mass-in-flow boundary conditions. Therefore, the segregation in the simulated system is mainly due to discharge and becomes significant toward the end of the discharge process.

(e) Discharge hopper with polydisperse particles in a layered configuration:

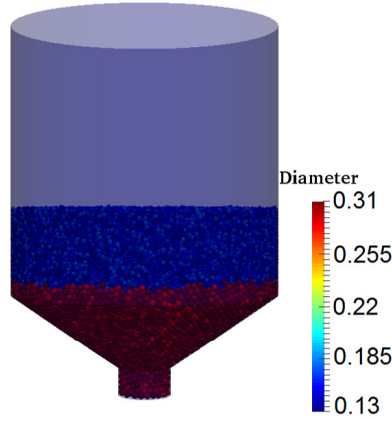


Figure 26. Initial configuration of the hopper discharge containing a layered packing of spherical beads with a bi-modal size distribution. The unit of the particle diameter is centimeter.

For the second validation case, we consider the discharge hopper with a layered configuration (see Fig. 26). The initial configuration for the discharge simulation is obtained by first filling the hopper with coarse beads from the top through the mass-in-flow boundary condition, while the bottom is kept closed through the no-flow boundary condition, and then continuing to fill the hopper with fine beads, which pack on top of the coarse beads (see Fig. 26). The injection velocities of the fine and coarse beads through the top are 16 cm/s, with a volume fraction of 0.02 for each layer. The resulting packing is allowed to settle to completely dissipate the kinetic energy, and the resulting layered stable packing is used as the initial start for the discharge simulation until the hopper is fully discharged. The number of computational grids used in this simulation is 30 by 30 by 40 respectively along the two lateral directions and the axial direction of the hopper. The domain decomposition configu-

ration is 4 by 4 by 4 with 64 cores and the total CPU hours for the discharge simulation are 6.81 hours.

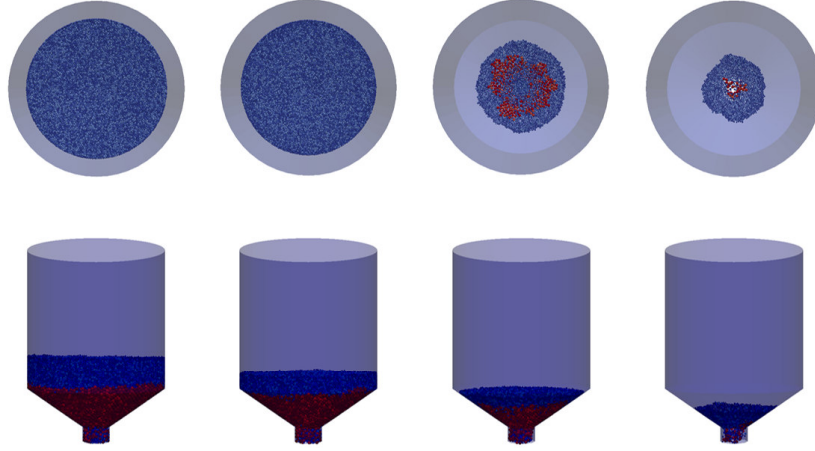


Figure 27. Snapshots of the discharge simulation of the layered configuration at 6 s, 9 s, 11 s and 12 s, respectively, from left to right. The upper panels show the top view and the lower panels show the side view.

Fig. 27 shows the snapshots of the discharge simulations. The discharge dynamics as quantified via $\gamma_f^N - \bar{\gamma}$ are shown in Fig. 28. As seen in Fig. 27, in this case, the coarse particles packed at the hopper bottom discharge first, which opens up a channel for the fine particles on the top. Therefore, the resulting γ_f^N remains zero for a while (before a channel is opened up), and is monotonically increasing as the discharge proceeds after the channel is open. Towards the end of discharge, since most of the fine particles in the hopper are gone, γ_f^N begins to decrease and finally starts to fluctuate due to a very small number of fine particles remaining in the hopper. Again, our simulated $\gamma_f^N - \bar{\gamma}$ curve agrees very well quantitatively with the experimental data, as shown in Fig. 28.

Our simulation also reveals how the channel for fine particles is opened up during the discharge. Due to the geometry of the hopper, the particles in the central region corresponding to a virtual extension of the cylinder-shaped outlet at the hopper bottom always discharge first, regardless of the initial packing configurations in the hopper. The fast flow of discharging particles in this central region generates a pressure on the remaining particles and pushes them against the container wall. In the layered configuration, the coarse particles are packed at the bottom of the hopper and fine particles are then stacked on top. Thus, once the discharge process starts, the coarse particles in the central region discharge first, which efficiently open up a channel for the fine particles that were originally packed on top of the coarse ones.

Again, it can be seen that the experimental data tends to deviate more from the ideally mixed case (i.e., unity) compared to the simulation results. Although in this layered config-

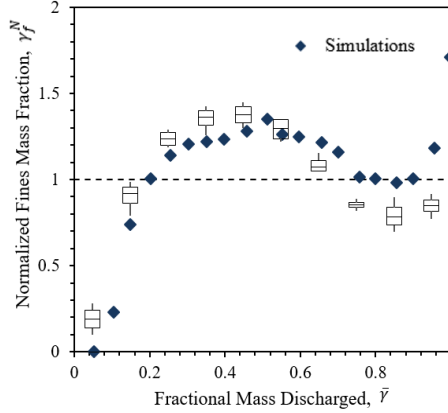


Figure 28. Discharge dynamics are quantified using the normalized fine mass fraction γ_f^N of the discharged particles vs. the overall discharged mass fraction $\bar{\gamma}$ for the layered configuration. The experimental results are represented as a box plot.

uration, the coarse and fine particles are separately packed in their own layers, within each layer the locally segregated regions still exist (due to the polydispersity of particles within the same solid phase) in the experimentally prepared initial configurations. This leads to the observed stronger segregation effects during the discharge process. For better controlled initial configurations in the simulations, the segregation is better suppressed and thus, the deviation of the simulated results from the ideally mixed case is less significant.

3.1.3 - Particle Height Classification Problem

(a) Background and motivation:

In many industrial applications involving chemical reaction, the final distribution of solid particles with different chemical composition after the reaction can contain important information related to the reaction efficiency etc. However, the full 3D packing configuration of particles after reaction is typically not available as these particles are stored in closed containers such as a hopper. After the reaction, the particles are discharged and typically collected in a train of buckets. By analyzing the discharged particles in each individual buckets, one can get estimates on certain discharge dynamics, e.g. the resistance time of the particles, as well as the particle characteristics such as size and density. It is then of great interest and importance if one could inversely reconstruct the particle packing configuration from such limited information on discharge dynamics.

Motivated by the needs in reaction design and optimization, we develop here an inverse reconstruction procedure that enables one to obtain the particle height information in the

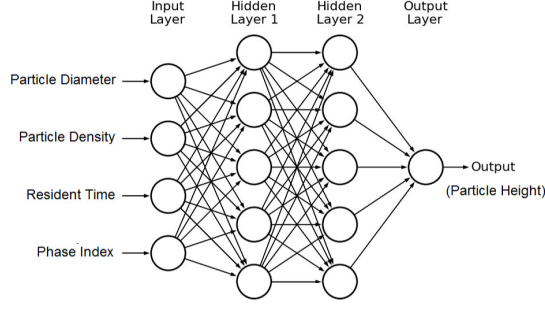


Figure 29. Schematic illustration of Convolutional Neural Network (CNN) model used for particle height classification.

initial packing configuration in the hopper bin from limited discharge dynamics data and particle characteristics. Our procedure is based on convolutional neural network (CNN) models, which take experimentally measurable particle resistance time, radius and density as input, and provide a classification of particle height in the initial packing as output. Using MFI-X-DEM simulations with enhanced physical modeling capabilities, we generate extensive discharge data for hoppers containing two distinct solid phase particles with four distinct classes of initial packing geometries. The CNN reconstruction models are subsequently trained and tested using the discharge data.

(b) Formulation of the inverse particle height classification problem:

In our problem, we pre-partition the hopper into N_b bins with uniform width s and index $\alpha = 1, \dots, N_b$. In this way, the particle height reconstruction problem is reduced to a classification problem, in which a particle is assigned a bin number α based on its resident time t_r , diameter D_p and density ρ . We note that in this work we only utilize particle height information to quantify its original position in the hopper. This is due to symmetry of initial packing configurations of interest to us, e.g., a layered configuration with multiple solid phases. In principle, one can also include radial information besides particle height for more complex initial packings. However, this would add additional input data and a more sophisticated reconstruction model, which is out of the scope of the current work.

We now present our Convolutional Neural Network (CNN) model for this reconstruction (height classification) problem. We employ CNN method instead of other techniques mainly because of its proven superior performance for handling very rough optimization landscape, which is the case for our problem due to the large dispersion in the resident time (see the section below). The CNN reconstruction model contains 4 layers, including an input layer, 2 hidden layers, and an output layer. The number of neurons in each layer is respectively 500, 500, 500, and 15 neurons in each layer respectively. The input data includes the set (t_r, D_p, ρ) for each particle, extracted from the discharge simulation data. The output information is the index α of the bin that the particle falls into. A dropout rate of 0.3 is used in case of over-fitting. The activation functions for each layer used are respectively: *relu*, *relu*, *relu*, and

softmax. The CNN network is developed and implemented based on the *TensorFlow* and *Keras* packages employing python scripting language, which are both open-source software [3, 4, 9]. We have also experimented with our network set-ups by varying the number of hidden layers and the number of neurons in each layer. The current set-ups result in the best reconstruction accuracy for the initial packings that we investigate.

In order to quantify the goodness of the reconstruction (height classification), we introduce an accuracy metric M . Specifically, once a bin classification $\alpha \in [1, \dots, N_b]$ is obtained, the height of the particle i can be estimated as

$$h_i = (\alpha - 0.5) \times s, \quad (2.3)$$

where i is the bin index and s is bin width. We then define the success function I , i.e.,

$$I_i = \begin{cases} 1 & |h_i - h_i^*| \leq \sigma, \\ 0 & |h_i - h_i^*| > \sigma, \end{cases} \quad (2.4)$$

where h_i^* is the actual height of particle i , and σ is the tolerance value. The accuracy metric M is then defined as the average of success function values for all testing particles, i.e.,

$$M = \frac{1}{N_t} \sum_{i=1}^{N_t} I_i. \quad (2.5)$$

In the subsection below, we will present the reconstruction accuracy for the CNN models for a wide spectrum of distinct hopper packings.

(c) Results of the height classification study:

We present the discharge simulation data for hoppers containing two distinct solid phases with distinct initial packing geometries. Both solid phases consist of particles with a uniform size distribution, i.e., $D_p \in [0.05, 0.15]$ cm, where D_p is the diameter of particles. However, the two phases possess different densities with $\rho_1 = 2.5g/cm^3$ and $\rho_2 = 7.5g/cm^3$. The initial packing configurations are generated by first filling the hopper with $N = 200,000$ particles of solid phase # 1 using mass-in-flow boundary conditions. Then a sub-region of this initial packing with a specific geometrical shape is selected and all of the particles within this sub-region are designated as phase # 2 with ρ_2 . The newly constructed two-phase initial packing is then mechanically equilibrated for 10s before the discharge simulation.

In this work, we consider four classes of initial packing geometries, in which the denser particles of phase # 2 are respectively packed in a spherical region with radius R , a cuboidal region with height H and square edge length L , an upward cone region with height H and basal radius R , and a downward cone region with height H and basal radius R . Figure 30 schematically illustrates these four distinct packing geometries. For the sphere geometries, we consider $R = 3.5, 4.0, 4.5$ and 5.0 cm. For the cuboid geometries, we fix the height

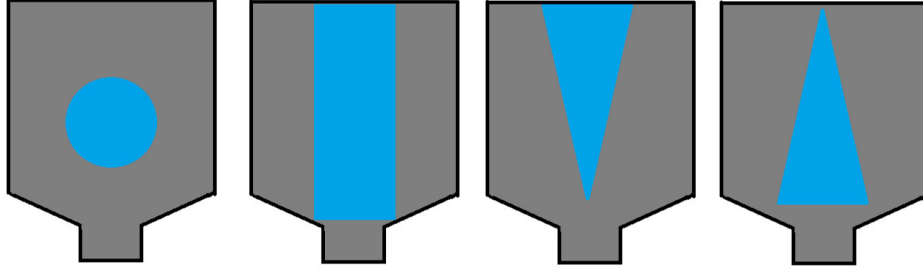


Figure 30. Schematic illustration of four distinct initial packing geometries for two solid phases in a hopper bin. The blue region represents the denser phase (i.e., phase # 2) and the gray region represents phase # 1.

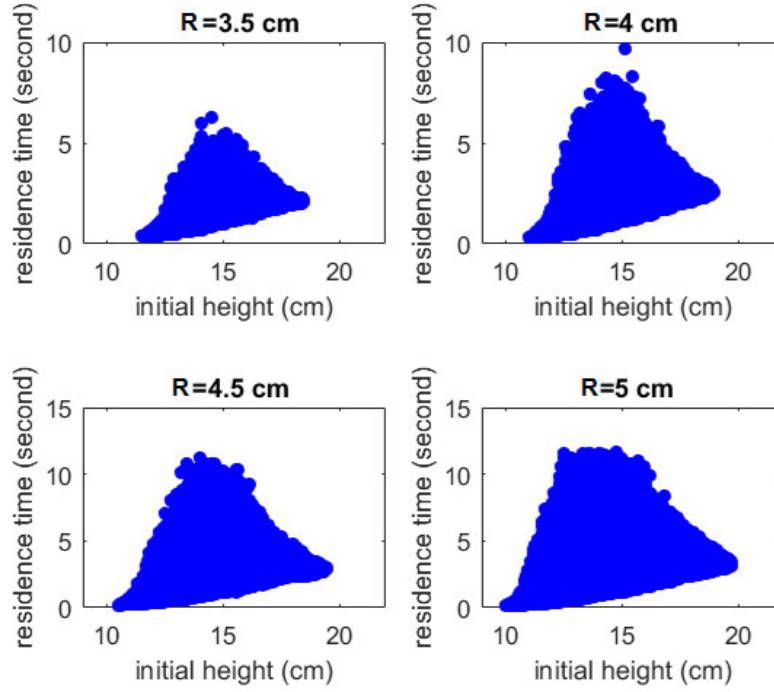


Figure 31. Scatter plots of resident time vs. particle height in hopper bin for the sphere geometries. The radii of the spherical region containing phase # 2 particles are respectively $R = 3.5, 4.0, 4.5$ and 5.0 cm.

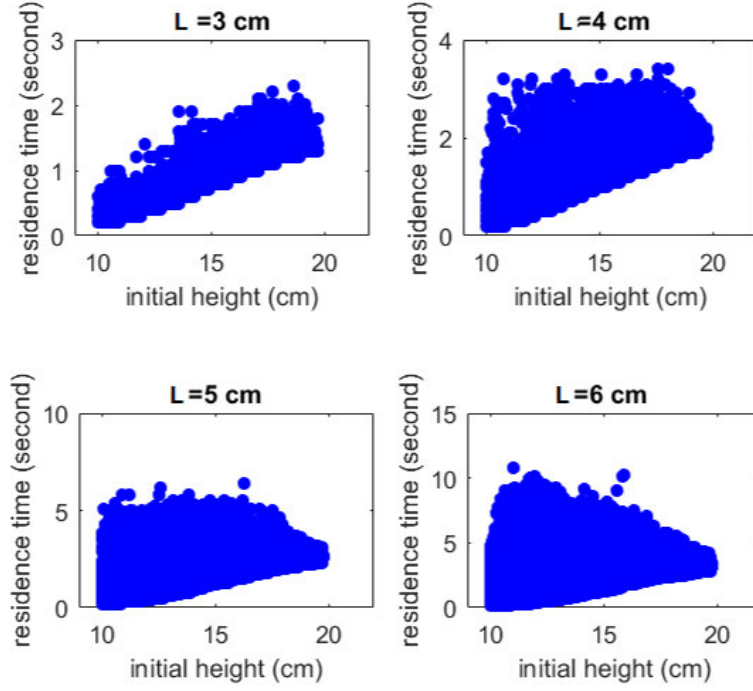


Figure 32. Scatter plots of resident time vs. particle height in hopper bin for the cuboid geometries. The edge lengths of the square basal plane of cuboid containing phase # 2 particles are respectively $L = 3.0, 4.0, 5.0$ and 6.0 cm. The height of the cuboidal region is fixed with $H = 12.0$ cm.

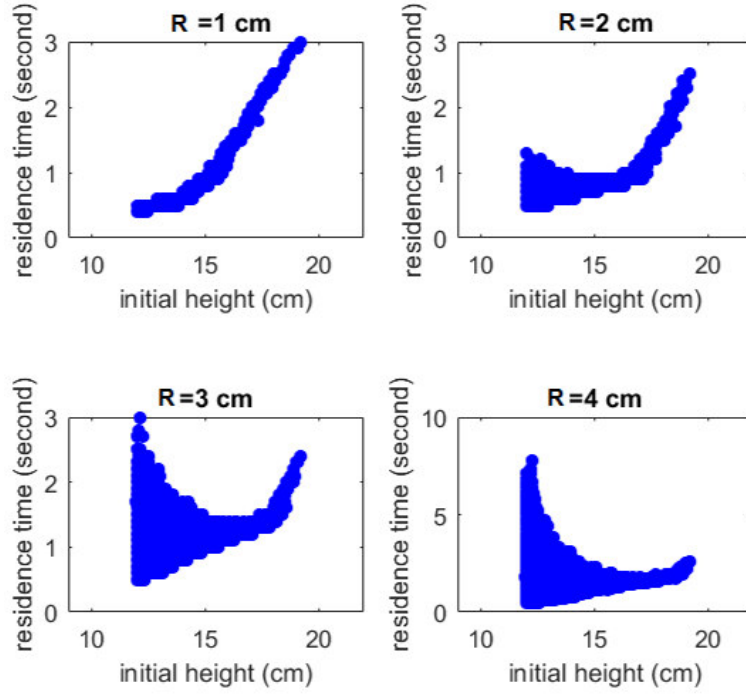


Figure 33. Scatter plots of resident time vs. particle height in hopper bin for the upward cone geometries. The radii of the basal plane of cone containing phase # 2 particles are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm. The height of the cuboidal region is fixed with $H = 10.0$ cm.

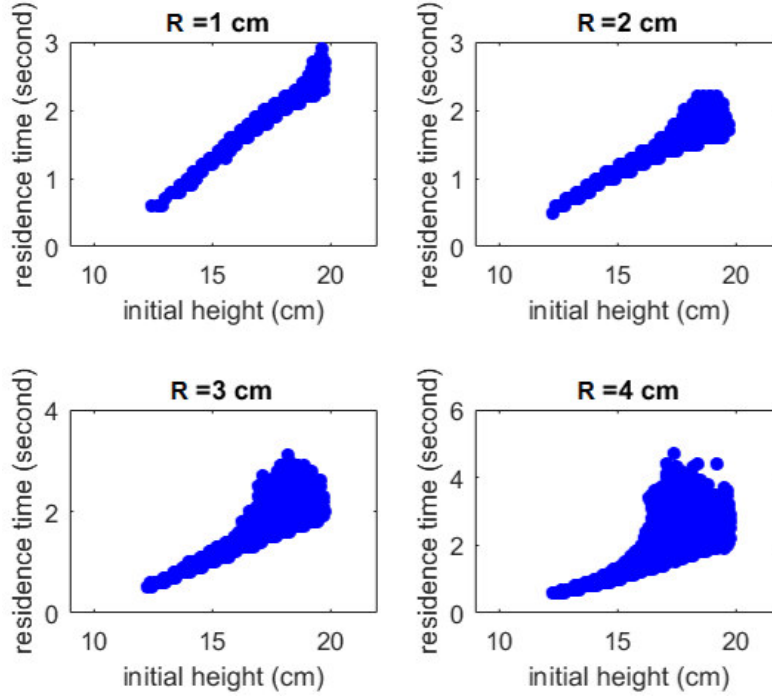


Figure 34. Scatter plots of resident time vs. particle height in hopper bin for the downward cone geometries. The radii of the basal plane of cone containing phase # 2 particles are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm. The height of the cuboidal region is fixed with $H = 10.0$ cm.

$H = 12.0$ cm and consider $L = 3.0, 4.0, 5.0$ and 6.0 cm. For both upward and downward cone geometries, we fix the height $H = 10.0$ cm and consider $R = 1.0, 2.0, 3.0$ and 4.0 cm.

Figures 31-34 respectively show the scatter plots of the resident time t_r (i.e., the total time that a particle stays in the hopper) vs. particle height for the four classes of initial packing geometries. It can be seen that for all cases, a specific particle height is generally associated with a range of resident time and vice versa. This implies that in general it is not possible to exactly locate the original particle height from the associated resident time alone. As we will show below, the accuracy of the inverse reconstruction (height classification) is affected by the degree of dispersion of the resident time for a given particle height. The above discharge data, together with the particle characteristics including the diameter and density, will be subsequently used as input data to train the CNN reconstruction models.

For the two-phase hopper configurations we consider, resolving the initial packing of one phase is sufficient to determine the overall packing configuration. Therefore, in the subsequent discussion, we will focus only on phase # 2 (i.e., the denser particles) and use CNN models to estimate the height of these particles in the initial packing.

The discharge simulation data are employed to train three types of CNN reconstruction models, including hopper-specific models, geometry-specific models and a generic model. For the hopper specific models, we only use the discharge data for a specific hopper configuration to train the CNN model. In particular, the resident time data associated with 70% randomly selected phase-2 particles are used as training data for establishing the CNN model, and the data for the remaining 30% phase-2 particles are used as testing data. A geometry-specific CNN model is trained using discharge data for hoppers with the same class of initial packing geometry. For each geometry, the data for one of the four cases are reserved as the testing data and the data associated with the remaining three cases are used to train the CNN model. For each training case, resident time data associated with all of the phase-2 particles are used. The generic CNN model is trained using discharge data for all four different classes of geometries. For each class of geometry, three cases are used for training and the remaining case is used for testing.

Figures 35 to 38 respectively show the reconstruction (height classification) accuracy metric M (see Eq. (2.5) for definition) for the hopper-specific models for the different four classes of initial packing geometries. For each case, M is plotted as a function of the tolerance σ . In each figure three M - σ curves associated with different bin width values s are shown. For all cases shown, it can be clearly seen that the accuracy M monotonically increases with increasing σ , which is well expected. We emphasize that even the largest tolerance $\sigma = 1$ cm considered here is still relatively small compared to the overall hopper height (17.5 cm). In addition, it can be seen that the reconstruction accuracy is not very sensitive to the choice of bin width s .

We now examine each class of packing geometry in detail. For the sphere packing geometry, as the sphere radius increases, the reconstruction accuracy decreases. The highest accuracy $M = 68\%$ is achieved for $R = 3.5$ cm and the lowest accuracy $M = 46\%$ is achieved for $R = 5.0$ cm. This result is not surprising. As briefly discussed in Sec. 3.1., the recon-

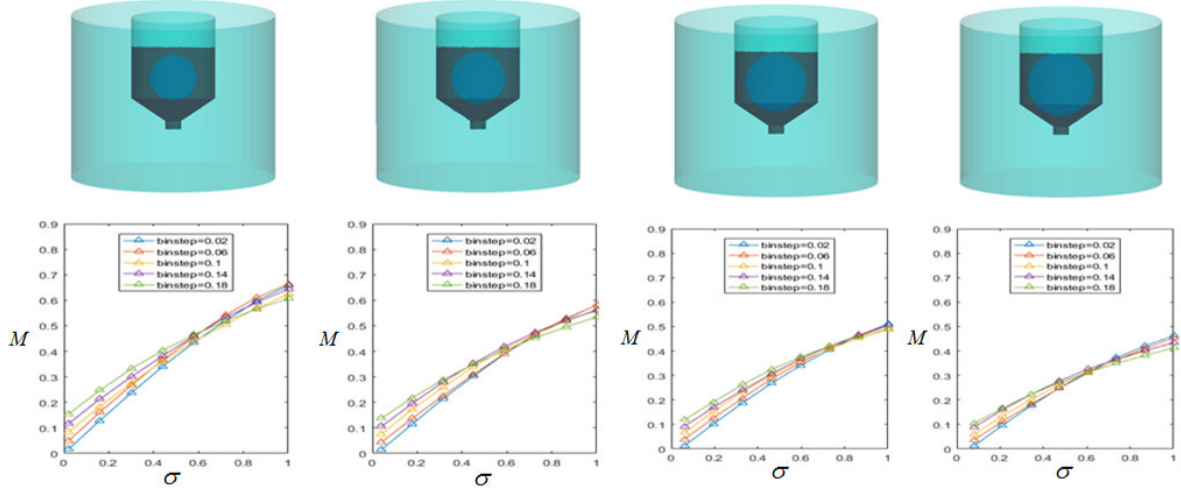


Figure 35. Upper panels: Snapshots of the initial packing configuration for the sphere geometry. The radii of the spherical region are respectively $R = 3.5, 4.0, 4.5$ and 5.0 cm from left to right. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.

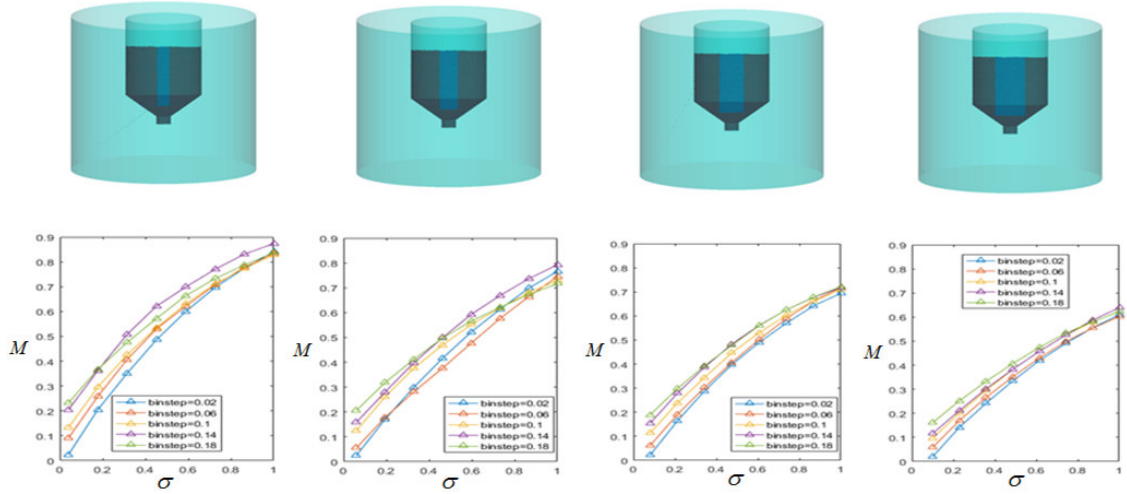


Figure 36. Upper panels: Snapshots of the initial packing configuration for the cuboid geometry. The edge lengths of the square basal plane are respectively $L = 3.0, 4.0, 5.0$ and 6.0 cm from left to right. The height of the cuboidal region is fixed with $H = 12.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.

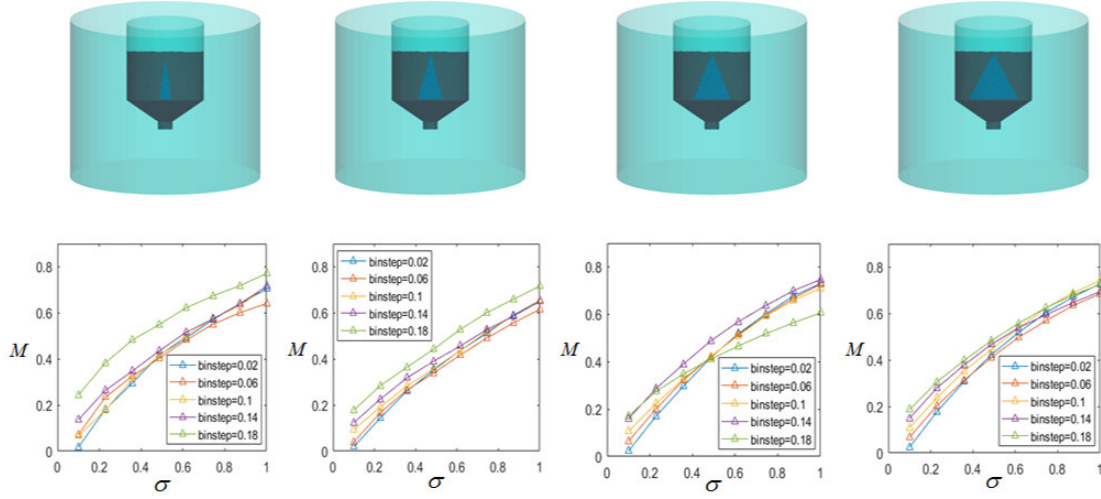


Figure 37. Upper panels: Snapshots of the initial packing configuration for the upward cone geometry. The radii of the basal plane are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm from left to right. The height of the cone is fixed with $H = 10.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.

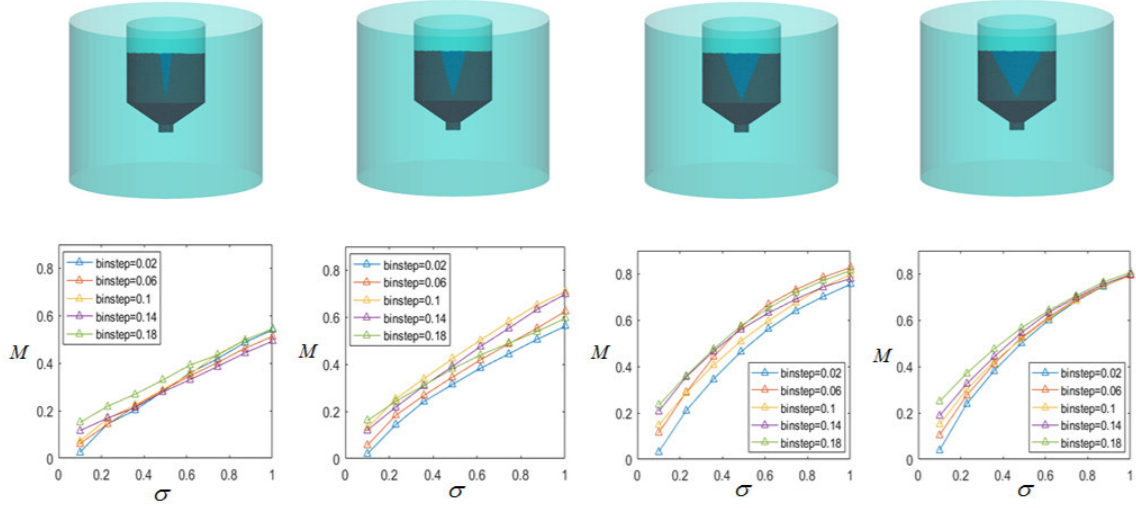


Figure 38. Upper panels: Snapshots of the initial packing configuration for the downward cone geometry. The radii of the basal plane are respectively $R = 1.0, 2.0, 3.0$ and 4.0 cm from left to right. The height of the cone is fixed with $H = 10.0$ cm. Lower panels: The reconstruction accuracy M as a function of the tolerance σ (in cm) for different bin width s for the hopper-specific CNN models.

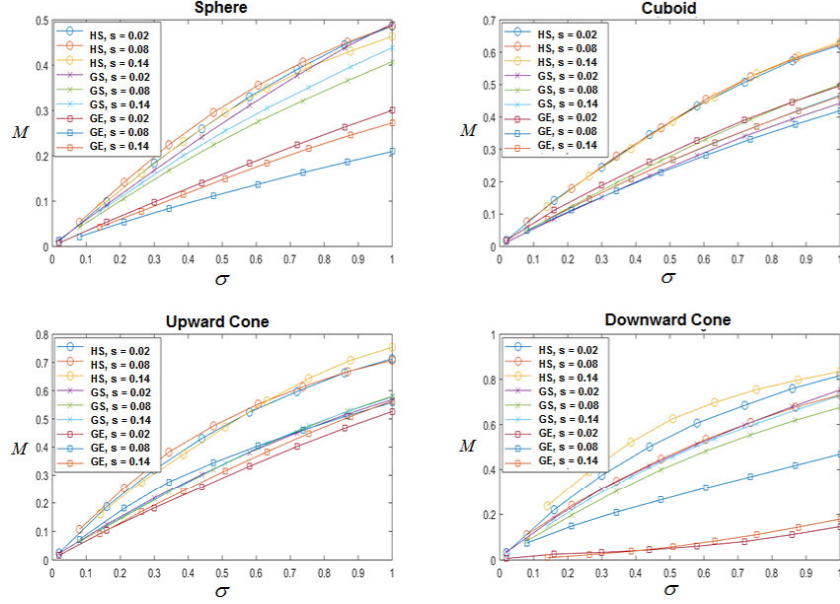


Figure 39. The reconstruction accuracy M as a function of the tolerance σ for different bin width s for the geometry-specific (GS) models and the generic model (GE), compared to the corresponding hopper-specific (HS) models. One packing configuration not used for training the models is chosen from each geometry class. The results shown respectively correspond to the sphere, cuboid, upward cone and downward cone geometries.

struction accuracy is affected by the degree of the dispersion of the resident time t_r for a given particle height. For sphere regions with larger R , a specific particle height can be associated with a wide range of t_r , which leads to worse reconstructions. For the cuboid and upward cone geometries, the trends are very similar to that in the sphere case, i.e., the accuracy decreases as the basal plane size increases. For the cuboid geometry, the highest $M = 89\%$ is achieved for $L = 3.0$ cm. For the upward cone geometry, the highest $M = 88\%$ is achieved for $R = 1.0$ cm. Interestingly, for the downward cone geometry, increasing the basal plane radius R leads to an increase of M , with the highest $M = 80\%$ is achieved for $R = 4.0$ cm. This is because the downward cone shape corresponds very well to the flow pattern during discharge, which results in small dispersion in t_r for a specific particle height. Since very specific discharge data are used to train the CNN models, it can be expected the obtained accuracy from these hopper-specific reconstructions can serve as benchmarks (e.g., upper bounds) for subsequent models.

Figure 39 shows the reconstruction accuracy of the geometry-specific models and the generic model, compared to that of the hopper-specific models. We choose one packing configuration from each geometry class, which has not been used for training the models. The resident time data as well as the particle characteristics are then used as input for the

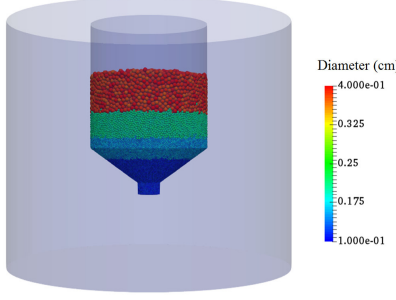


Figure 40. Snapshot of the initial packing configuration for the four-phase system in a layered configuration. The particles for phases #1, 2, 3 and 4 are respectively shown with dark blue, light blue, green and red colors.

CNN models for particle height reconstruction. From Fig. 39, it can be clearly seen that both the geometry-specific models and the generic model generally yield less accurate height classification results than the hopper-specific models, which is expected. For the sphere and downward cone packings, the geometry-specific models are significantly better than the generic model, and can lead to a M comparable to that of the hopper-specific models. For the cuboid and upward cone packings, the geometry-specific models and the generic model result in similar accuracy values. We note that these results are not dependent on the specific choice of training and testing cases.

We now apply the same procedure to reconstruct the particle height information for a hopper bin with four distinct solid phases to further demonstrate its utility. Different from the two-phase hopper systems, in this case the four solid phases possess the same density $\rho = 2.5g/cm^3$. However, each phase possesses a distinct normal distribution of particle sizes, with the average particle diameter $\bar{D}_p = 0.05$ cm, 0.08 cm, 0.10 cm and 0.19 cm respectively for phases # 1, 2, 3 and 4 and a standard deviation of 0.03 cm for all phases. The number of particles for phases # 1, 2, 3 and 4 is respectively 80,000, 60,000, 40,000 and 20,000. The initial packing of the particles possesses a layered geometry, see Fig. 40. This configuration is obtained as follows: the bottom outlet of the hopper is closed and the particles of phase # 1 are filled into the hopper via mass-in-flow (MIF) boundary condition and are packed at the bottom of the hopper. When a mechanically stable packing of phase # 1 particles is achieved, we begin to inject phase # 2 particles using MIF. Following the same procedure, we fill phase # 3 particles and phase # 4 particles.

Once the initial packing configuration is constructed, we perform the MFX-DEM simulation to obtain the required discharge data (i.e., the resident time t_r for each particle). Figure 41 shows the scatter plots of resident time vs. particle height for all four phases. It can be seen that due to the layered geometry, the dispersion of t_r for all phases is relatively small compared to the two-phase hopper cases. Similar to the hopper-specific models discussed

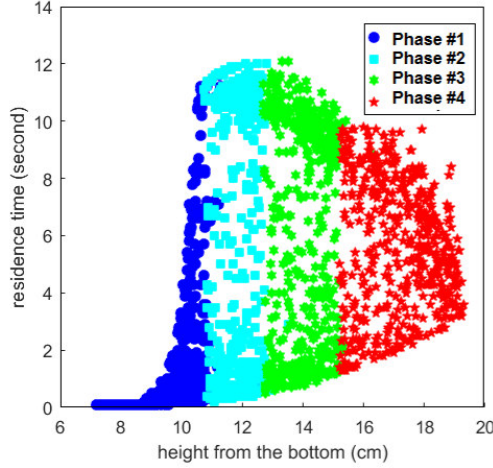


Figure 41. Scatter plots of resident time vs. particle height for each phases in the four-phase system. The data for phases #1, 2, 3 and 4 are respectively shown with dark blue, light blue, green and red colors.

in Sec. 3.2., for the four-phase system, the discharge data associated with 70% randomly selected particles for each phase together with the particle diameters and density are used to train the CNN model and the data for the remaining 30% particles are used as testing data.

Figure 42 shows the accuracy metric M as a function of σ for different bin width values s for all four phases. Consistent with the results for the hopper-specific models for the two-phase systems, it can be clearly seen that M is a monotonically increasing function of σ and is not sensitive to the choice of bin width s . Remarkably, a very high accuracy $M = 95\%$ can be achieved for phase #3 with a $\sigma = 0.8$ cm. This is again due to the very strong correlation between t_r and the initial particle height for this phase. The reconstruction accuracy for phases #2 and #3 is also relative high, i.e., larger than 90%. For #1, the accuracy is about 75%, which is due to the large dispersion in t_r associated with a specific particle height.

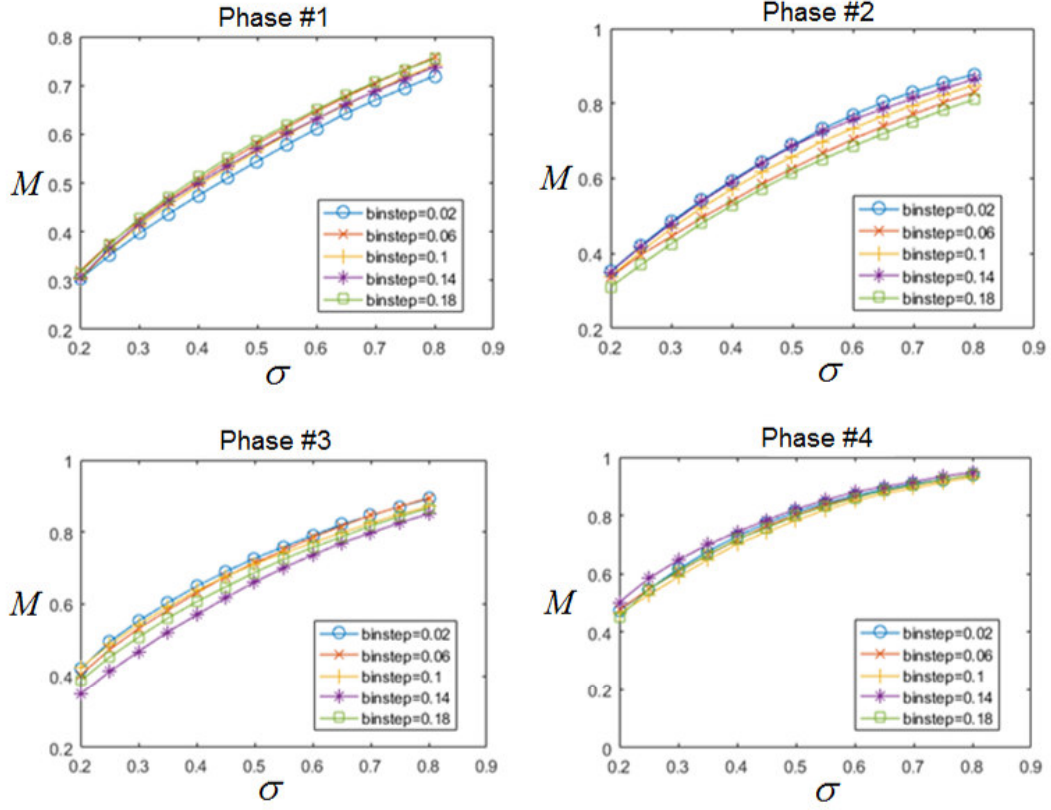


Figure 42. The reconstruction accuracy M as a function of the tolerance σ for different bin width s for each phase in the four-phase system. The results for phases #1, 2, 3 and 4 are respectively shown.

Subtask 3.2 - Validation of Heat Transfer Models

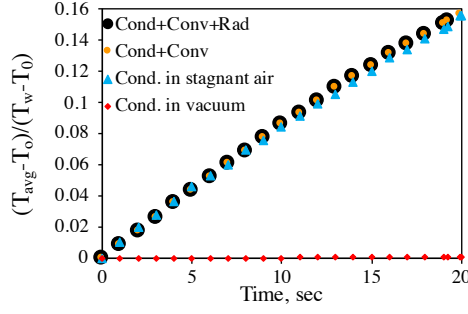
The objective of this subtask is to validate the existing heat transfer models in MFIX-DEM since these models had not been previously tested with more than two particles based on the available published literature information at the beginning of the project. Thus, we used a rotary drum filled with particles as our benchmark problem configuration to test heat transfer process, which is a widely used configuration in many industries. We tested all heat transfer modes (conduction, convection, and radiation) in MFIX-DEM using this configuration. Also, we have validated the conduction heat transfer model implemented in MFIX-DEM with our own simulations and EDEM simulations, as well as literature simulations and experimental data. We have also constructed a lab-scale instrument for local experimental capabilities and performed preliminary conduction experiments in support of the validation activities.

3.2.1 - MFIX-DEM Heat Transfer Verification and Validation

(a) Testing All Heat Transfer Modes:

MFIX-DEM has models implemented for all forms of heat transfer that can occur in a multiphase medium, including conduction, convection, and radiation. For this and all of our simulation studies, we used a rotary drum geometry filled with particles, which is a common equipment setup in many industries. In our setup, an aluminum drum of 15.24 cm diameter and 1.5 cm axial length is half-filled with monodispersed alumina particles of 0.2 cm diameter. A total of 20,000 particles are generated and allowed to fall under gravity while the drum is rotating. All of the particles are initially given a temperature of 298 K. Once the particles inside the drum reach mechanical equilibrium, the wall is instantaneously heated and maintained at a temperature of 398 K, and the drum is rotated at 20 rpm. The heated wall of the drum has the no-slip condition, and the front and back walls of the domain are free-slip adiabatic boundaries. The geometry is held fixed in time, and the rotation of the drum is generated by modifying the gravity vector. As the drum is rotated, the particles move due to friction between the wall and particles. At the same time, heat is transferred from the higher temperature wall to the lower temperature particles, and from the higher temperature particles to the lower temperature particles.

The results for the average temperature profile over time for each of the different heat transfer modes in the rotary drum is given in Fig. 43. Here, it can be seen that the heat transfer models appear to be functional, except no difference is seen with the radiation model, which may indicate the need to improve this mathematical model to better represent the phenomena.



		Conduction vacuum	Conduction in stagnant air	Convection & Conduction	Conduction, convection and radiation
Solid Phase	Specific heat [J/KgK]	840	840	840	840
	Thermal conductivity [W/mK]	1.05	1.05	1.05	1.05
	Thermal emissivity	0	0	0	0.8
Coupling	Drag model	-	-	SYAM_OBRIEN	SYAM_OBRIEN
Gas phase (air)	Specific heat [J/Kg-K]	0	1000.7	1000.7	1000.7
	Thermal conductivity [W/mK]	0	0.0261	0.0261	0.0261

Figure 43. Temperature distribution of monodispersed alumina obtained from MFIX-DEM simulations at (a) 1 s, (b) 5 s, (c) 10 s, and (d) 20 s.

(b) Conduction Verification: Further investigation was done for conduction to fully test this model under a variety of conditions. Table 3 summarizes the simulation parameters used for this investigation.

Figure 44 shows the temperature profile of the alumina bed at different time intervals. As time progresses, the near-wall particles heat up due to particle-wall contact, and these particles are transported to the freeboard as the drum rotates. The heat penetrates the solid bed to include more heated particles with subsequent rotations, thereby increasing the thermal boundary. Moreover, heated particles from near the wall transmit heat to surrounding particles as they move toward the freeboard region. This is seen after 10 s of simulation, see Fig. 44(c), where a thick layer of particles is heated along the boundaries of the solid bed. As time progresses towards 20 s, see Fig. 44(d), more particles are heated, forming a cool inner core and a warmer outer layer in the solid bed. However, a section of particles in the lower right side have a comparatively higher temperature than the surrounding particles, thereby creating a stagnant zone.

The average temperature is scaled with the wall temperature and the initial temperature of the particle bed to calculate the nondimensional temperature. Fig. 45 shows the evolution of temperature with respect to time obtained from the MFIX-DEM simulation compared with the DEM simulation published by Chaudhuri et al. [6]. The commercial DEM software, EDEM was used in their simulations, and they have validated their conduction heat transfer model using experiments. Both their simulation and experiment have predicted the same

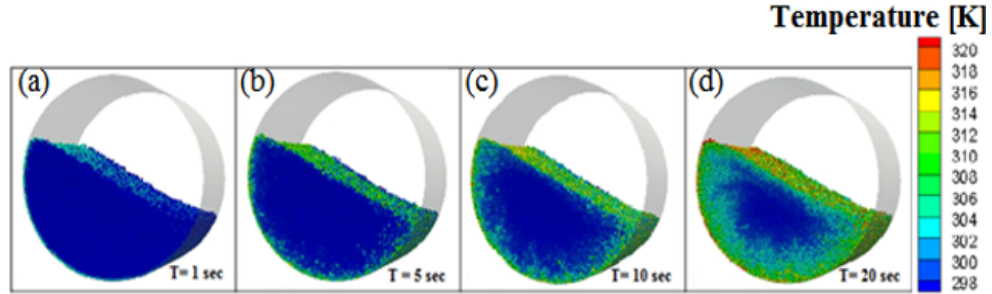


Figure 44. Temperature distribution of monodispersed alumina obtained from MFIX-DEM simulations at (a) 1 s, (b) 5 s, (c) 10 s, and (d) 20 s.

heat transfer profile, validating the model. Based on the credibility of the results presented, the current MFIX-DEM results were compared against their DEM results.

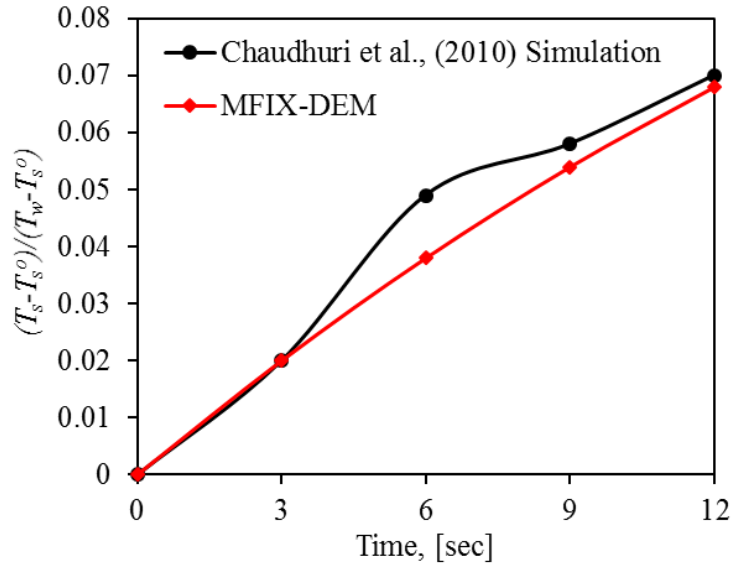


Figure 45. The evolution of average bed temperature for the MFIX-DEM simulation and Chaudhuri et al. (2010) DEM simulation.

For a better quantification of thermal behavior between the results, the thermal time constant is calculated using the slope in Fig. 46. The value of thermal time constant for 161.2 s is obtained from the literature simulation and 185.1 s from the MFIX-DEM simulation, which gives a percent difference of 13.7 %. This reflects the difference between two DEM simulation codes for mostly similar conditions as input, and more detailed research is required to understand the discrepancy. Therefore, in the subsequent sections, we will carry out a systematic parametric study to understand the sources of the discrepancies and their quantitative assessment. However, the average bed temperatures for both cases follow nearly an identical upward trend, with a maximum discrepancy of 13.7 % in the thermal time constant, indicating that the MFIX-DEM results agree fairly well with the DEM results presented by [6].

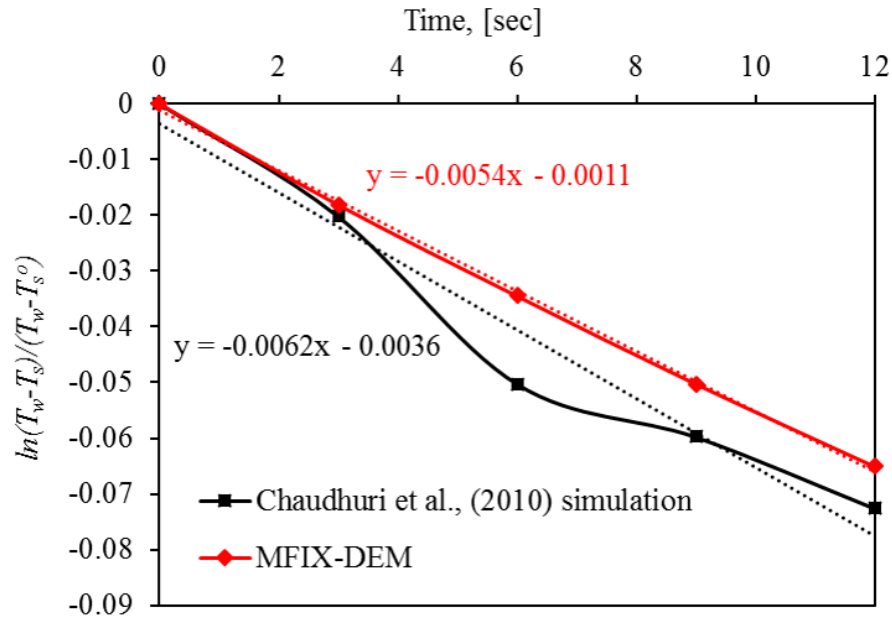


Figure 46. Logarithmic variation of scaled temperature difference with time for the MFIX-DEM simulation and Chaudhuri et al. [6] DEM simulation.

(c) Conduction Validation: The validation case is based on the experimental work performed by Chaudhuri et al. [6], which has similar conditions to their simulations (see Tab. 3), but with a 7.62 cm long drum. Fig. 47 shows the evolution of the scaled average bed temperature for both the MFIX-DEM simulation and experimental data available from the literature. The MFIX-DEM result shows the same increasing trend in the temperature as the literature experiment. To better quantify the accuracy of MFIX-DEM, the thermal time constant, is calculated using Fig. 48. The value of thermal time constant is 175.4 s for the literature experiment and 185.2 s for the MFIX-DEM simulation.

There are a few possible reasons for the 5.4 % discrepancy observed between the MFIX-DEM simulation and literature experiment. A sudden elevation in the temperature is ob-

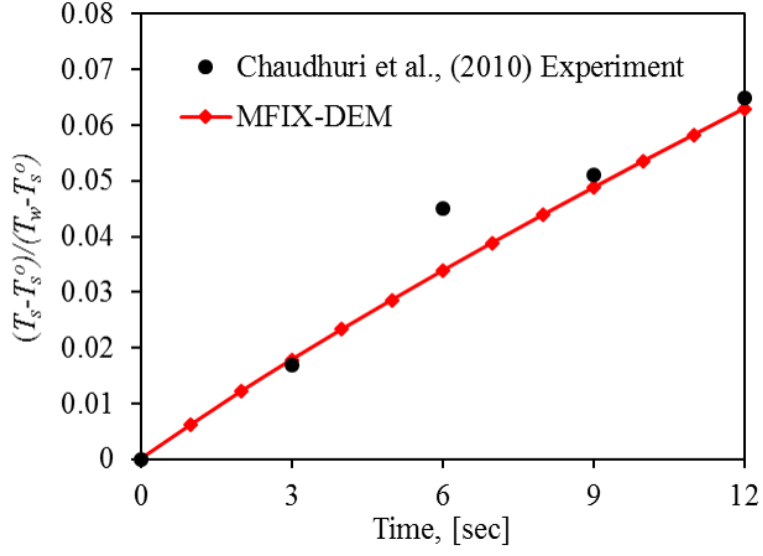


Figure 47. The evolution of the scaled average bed temperature for the MFI-X-DEM simulation and Chaudhuri et al. [6] experiment.

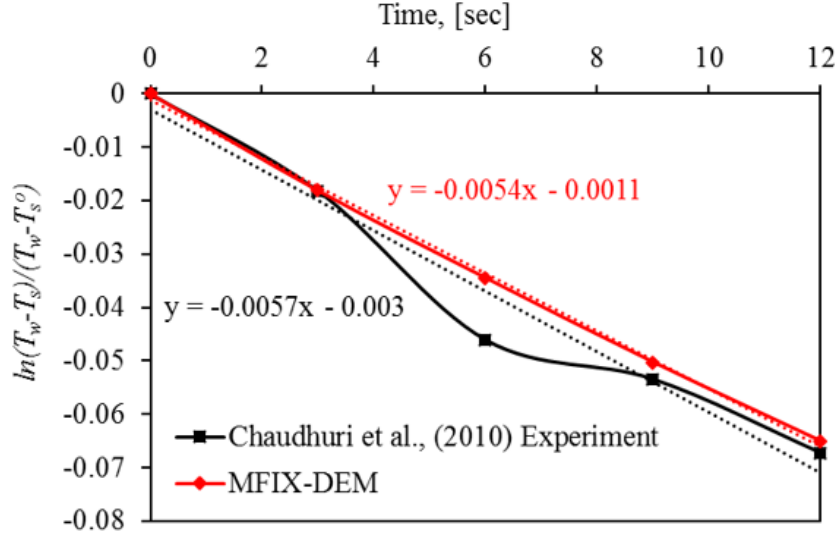


Figure 48. Logarithmic variation of scaled temperature difference with time for the MFI-X-DEM simulation and Chaudhuri et al. [6] experiment.

served at 6 s in the experiments, which was not explained by Chaudhuri et al.[6]. Due to the lack of confidence data in the experimental results reported, it is difficult to provide the exact reason for this discrepancy. For calculating the average temperature in experiments, the drum is stopped at 30 s intervals, then the thermocouples are inserted into the ten ports arranged radially along one of the side walls, and the average bed temperature is estimated as the mean of the readings of the ten thermocouples. When the drum is stopped for thermal measurements, it is likely that the temperature recorded by each thermocouple is an average measure of the interstitial fluid and particle temperatures. In such cases, the average temperature of the fluid- particle might read a greater value than the actual particle temperature, and result in a subsequent rise in the recorded temperature. Also, when the drum is stopped for recording the temperatures, the contact time between the solid bed and the wall increases, and simultaneously heat can be transferred to the particles, which might elevate the solid bed temperature. Another reason for the discrepancy can be due to particle size variations. Chaudhuri et al. [6] reported that the solid phase is monodispersed with 0.2 cm diameter alumina particles, and based on this, the current simulations are performed with single-phase monodispersed particles. But in reality, the experimental particles are most likely not perfectly monodispersed. Also, MFIX-DEM does not have the rolling friction capability implemented. In the literature, the rolling friction coefficient has been found to significantly affect the internal friction angle of dense granular processes. Therefore, it is important to investigate its effect on the hydrodynamic and thermal behavior of the particles in a rotary drum. In spite of the above discussed discrepancies, the MFIX-DEM conduction heat transfer model predicts the Chaudhuri et al. [6] experimental results with a discrepancy of 5.4 % in the thermal time constant. To better understand the reasons for the discrepancy, additional studies are performed to investigate the effects of model parameters on the contact heat transfer.

(d) Effects of Operating Parameters: After the initial literature experiment based validation of the conduction heat model, a parametric study is conducted by varying the rotation speed, particle size distribution, and rolling friction to better understand the effects of these operating parameters on the heat transfer model implemented in MFIX-DEM. A steel drum of 15 cm diameter and 2 cm axial length, filled with 0.11 kg of polydispersed alumina particles with 0.4 cm mean diameter and 0.33 relative standard deviation is used. All the particles are initially given a temperature of 298 K, and the drum wall is maintained at a temperature of 578 K. All the drum operating conditions and the simulation parameters are adopted from the DEM simulations performed by Emady et al.[12] (see Tab. 3), which employed the same conduction heat transfer model that was previously validated by Chaudhuri et al. (2010). The work published by Emady et al.[12] used polydispersed alumina particles as the solid phase for studying the conduction heat transfer in a rotary drum. Since one of our parameters of interest is particle size distribution, we have chosen to adopt the model parameters from this work.

Table 3. Simulation parameters used for verifying the effect of rotation speed on the contact conduction model in MFI-X-DEM with polydispersed alumina particles.

	Variable	Value
Global	Coefficient of restitution	0.2
	Time step	1.0×10^{-6} s
Particles	Material	Alumina
	Density	3890 kg/m ³
	Poisson's ratio	0.21
	Shear modulus	1.0×10^6 Pa
	Heat capacity, C_p	880 J/kgK
	Thermal conductivity, k	3000 W/mK
	Initial temperature, T_s^o	298 K
	Total particle mass	0.11 kg
	Particle radius, μ	0.2 cm
	Particle size distribution, σ/μ	Normal, 0.33
Drum	Material	Steel
	Poisson's ratio	0.3
	Shear modulus	7.93×10^7 Pa
	Wall temperature, T_w	578 K
	Diameter	15 cm
	Length	2 cm
	Average fill level	13 %
	Rotation speed, ω	5, 10, 20, 30 rpm

(e) Effect of Rotation Speed: Polydispersed alumina particles are heated at varying rotation speeds of 5, 10, 20, and 30 rpm. The effect of rotation speed, on the overall heat transfer coefficient, α for the alumina is shown in Fig. 49. An increase in α when rotation speed increased from 5 to 10 rpm, and a decrease from 10 to 30 rpm is observed. Fig. 50 shows the temperature profile of the solid bed for all rotation speeds, at $\tau/2$. It is expected that as the rotational speed increases, the contact time between the covered drum wall surface and solid bed decreases, thereby decreasing the heat penetration resistance. Also, higher rotation speeds promote better mixing, which results in a more homogeneous distribution of heat flux between the drum wall and solid bed. Thus, with low penetration resistance and better heat distribution at high rotation speeds, a greater overall heat transfer is expected. However, from Fig. 49, it is seen that the rotation rate does not seem to play a significant role on the contact heat transfer. A similar effect was observed by Emady et al. (2016), in their EDEM simulations (see Fig. 49). However, Emady et al. (2016) considered rolling friction in their studies, which might be one of the root causes of the discrepancy observed.

(f) Effect of Particle Size Distribution (PSD): MFI-X-DEM randomly generates particles using the user defined mean (μ), maximum, minimum and standard deviation (σ). Three series of normal distributions (batch 1, batch 2, and batch 3) are generated using 0.33 cm as the minimum, 0.46 cm as the maximum, 0.4 cm as the mean, and 0.035 cm as the standard deviation of the diameter. The PSDs generated in the three series are shown in Fig. 51, with a total of 867 particles each. The particles are given the properties of alumina (see Tab. 3) and the simulations are conducted at rotation speeds of 5, 10, 20, and 30 rpm. Fig. 52 shows the effect of rotation speed, ω , on the overall heat transfer coefficient, α for the three batches. As observed in the previous section, rotation speed has no significant effect on the contact heat transfer. At lower rotation speeds, an increase in heat transfer is

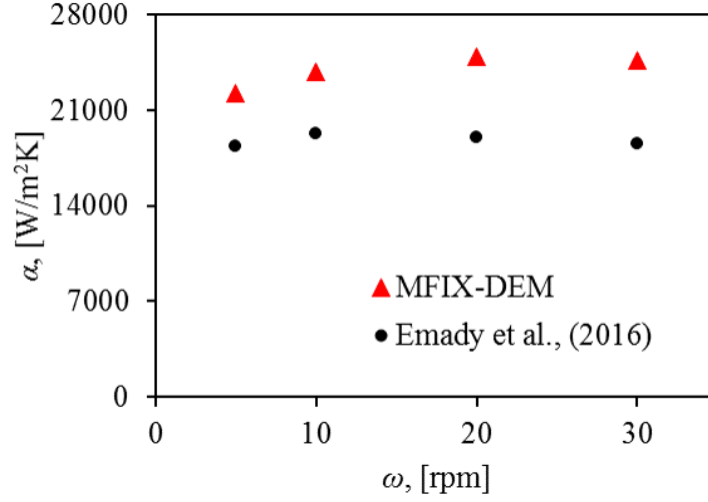


Figure 49. The sensitivity of the overall heat transfer coefficient to rotation speed for the MFIX-DEM simulations and Emady et al. (2016) DEM simulations.

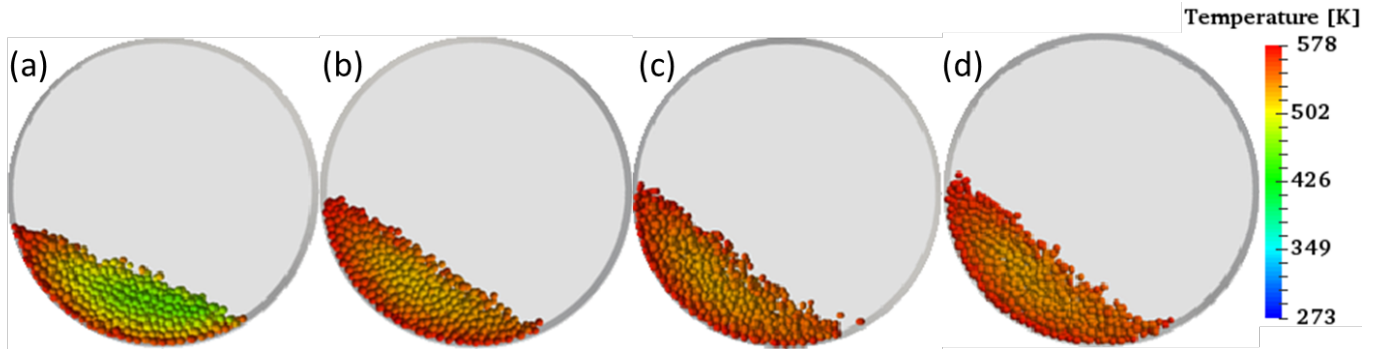


Figure 50. Temperature distribution of polydispersed alumina, obtained from MFIX-DEM simulations at $t=2$ for rotation speeds of (a) 5 rpm, (b) 10 rpm, (c) 20 rpm, and (d) 30 rpm.

observed along with rotation rate, whereas at higher rotation speeds, as the rate increases a decrease in heat transfer is observed.

The overall heat transfer coefficient, α , varied with a standard deviation of 267.1, 431.4,

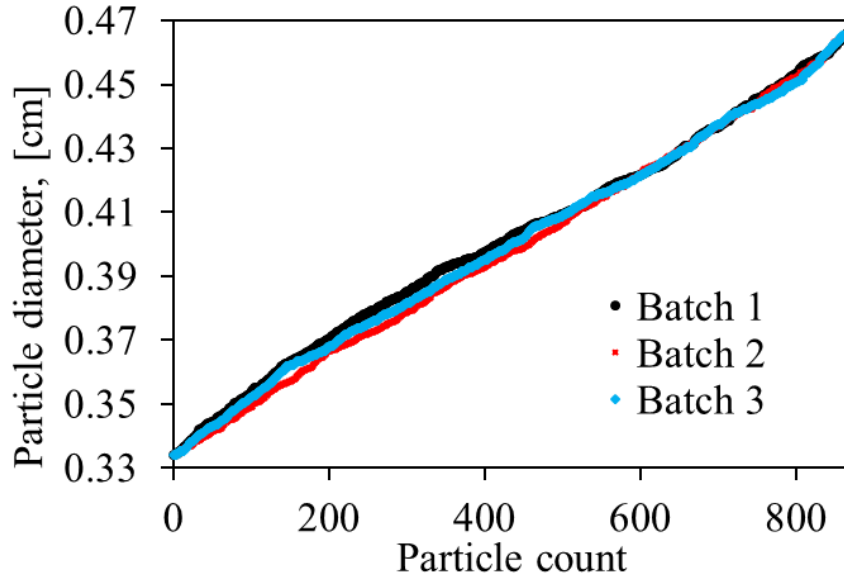


Figure 51. Particle size distributions with $\mu = 0.4$ cm and $\sigma = 0.035$ cm.

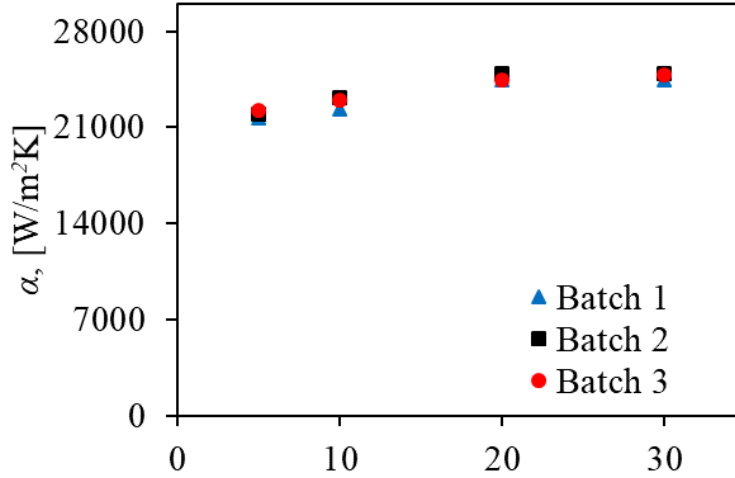


Figure 52. The sensitivity of the overall heat transfer coefficient to rotation speed for three PSDs with $\mu = 0.4$ cm and $\sigma = 0.035$ cm.

247.9, and 213.9 W/m²K at rotation speeds of 5, 10, 20, and 30 rpm, respectively, which contributes up to approximately 2 % of the observed values. Therefore, while performing validation studies using polydispersed particles, a certain percentage of discrepancy observed between the results might be caused by the failure to replicate the particle size distribution in the experiments. Hence, the 5.4 % discrepancy observed while validating the contact heat transfer model in MFIx-DEM might have a certain percentage contributed by the difference

in PSD. Thus, additional simulations are performed to study the effect of distribution range. Three simulations are performed using different types of distributions, namely, homogeneous, narrow, and broad. Fig. 53 shows the particle distribution obtained from MFIX-DEM for the three distributions. All three distributions have a mean diameter value of 0.4 cm. The narrow and broad distributions have a range from 0.34 to 0.47 cm and 0.21 to 0.6 cm, respectively (see Tab. 4). These values have been chosen to keep the difference in PSD minimal and observe its effect. An overall heat transfer coefficient of 52, 50, and 48 kW/m²K is found for the homogeneous, narrow, and broad distributions, respectively (see Tab. 4). A percentage difference of 4 % is observed between the homogeneous and narrow, and 9 % between the homogeneous and broad distributions. It can be observed that the case with a broad distribution had a lower heat transfer rate, and the homogeneous distribution case had a higher heat transfer rate. Since polydispersed particles exhibit segregation, the conduction heat transfer is inhibited.

Table 4. Particle size distribution information used for the narrow, homogeneous, and broad distributions.

Type of distribution	Diameter, cm				Number of particles	α , kW/m ² K
	Minimum	Maximum	STDEV	Mean		
Homogeneous	-	-	-	0.4	396	52
Narrow	0.34	0.47	0.03	0.4	400	50
Broad	0.21	0.6	0.07	0.4	398	48

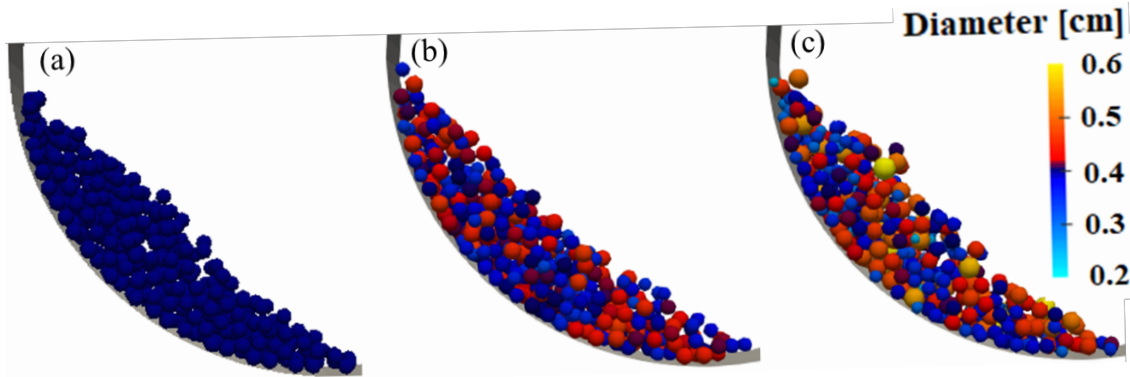


Figure 53. Particle size distributions obtained from MFIX-DEM simulations at $t=\tau/2$ for (a) homogeneous, (b) narrow, and (c) broad distributions.

(g) Effect of Rolling Friction: Since MFIX-DEM does not have a rolling friction model implemented, to assess the effect of this feature, we performed additional simulations with

the commercial code EDEM, which has rolling friction, and compared the results using both codes for the same problems and configurations. The force models and heat transfer models used in these simulations are listed in Emady et al.[12]. Simulations are performed using rolling frictions of 0, 0.005, 0.01, and 0.1. These values are selected based on previous DEM rotary drum simulation works. Based on our knowledge, most researchers have been using a rolling friction value of less than 0.1 in their studies. Specifically, while studying spherical particles, where the surface is considered to be smooth, a lower value of rolling friction is expected to yield more realistic behavior. Therefore, we have varied the rolling friction value from 0.005 to 0.1. Fig. 54 shows the effect of rolling friction on the overall heat transfer coefficient, α , at rotation speeds of 5, 10, 20, and 30 rpm. It can be seen that with increasing rolling friction, the heat transfer rate decreases. Comparing with the no rolling case at 5, 10, 20, and 30 rpm, a percentage change in the overall heat transfer coefficient of 2 %, 3 %, 1 %, and -4 % is observed for 0.005 rolling friction; 8 %, 5 %, 1 %, and -2 % is observed for 0.01 rolling friction; and 32 %, 42 %, 26 %, and 28 % is observed for 0.1 rolling friction, respectively. For instance, at 10 rpm, a higher heat transfer rate is observed for the no rolling case, and a lower heat transfer rate is observed at a rolling friction of 0.1. From 0 to 0.1, there is a 200 % change in rolling friction, and the change in heat transfer coefficient is 42 %, which shows that rolling friction has an effect on the heat transfer between the drum walls and the process material. At lower values of rolling friction, particle circulation in the bed is enhanced and therefore a better heat transfer is achieved.

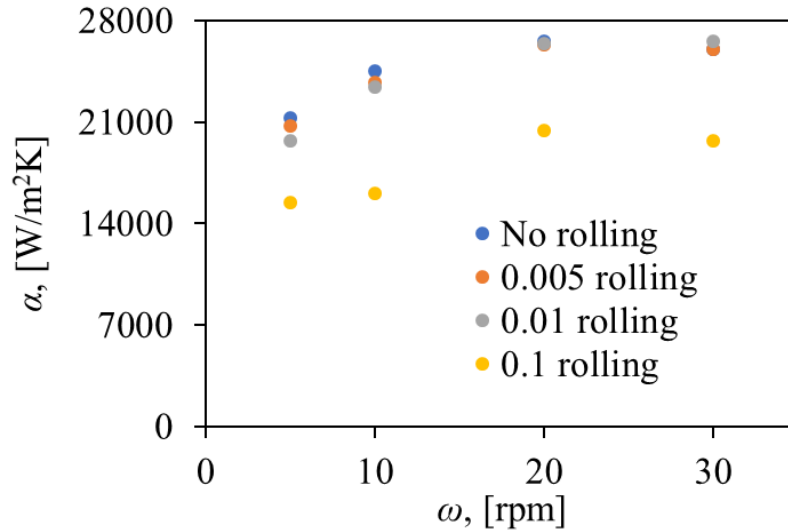


Figure 54. The sensitivity of the overall heat transfer coefficient to rotation speed at rolling frictions of 0, 0.005, 0.01, and 0.1.

(h) Heat Transfer Local Experiments: We constructed a local experimental setup for rotary drum heat transfer. The goal was to validate the MFI-X-DEM simulations with our own experiments (see Fig. 55). However, the construction of the drum took longer than anticipated, but we were still able to carry out a few experiments as part of the Phase 1

project activities. Here, a design of experiments was performed where the fill level was varied at 4 %, 7 %, and 10 %, and the rotation rate was varied at 1, 5, and 9 rpm. The average bed temperature after an hour of experiment was measured for each run. Our results showed that fill level has a significant effect on the heat transfer rate, where heat transfer increases at lower fill levels. Also, rotation rate was found to have a minimal effect, which is consistent with our MFIX-DEM simulation findings.

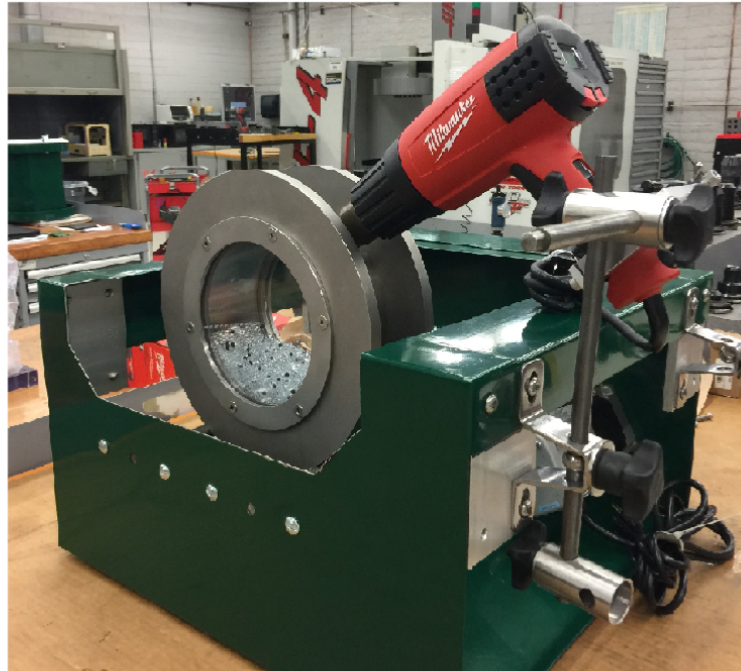


Figure 55. Local experimental capabilities for rotary drum heat transfer for MFIX-DEM validation.

Conclusions and Recommendations for MFIX-DEM Heat Transfer:

Conduction heat transfer has been validated with literature data, and the effects of operating parameters were studied with simulations and experiments. The main findings were that: (1) Rotation rate has no significant effect, and (2) PSD and rolling friction have significant effects. It is recommended that MFIX implement rolling friction. The next step for heat transfer is to validate the models for convection and radiation.

Task 4 - Ascertain Predictive Credibility in MFIX-DEM

With major advances in mathematical modeling and high performance computing, simulation-based engineering has become one of the three pillars (the other two are theory and experiments) in modern scientific discovery and engineering system analysis. The translation from physical reality to a set of mathematical equations and finally to simulation codes, however, introduces many errors and uncertainties, which adversely affect the credibility of simulation predictions. To mitigate these difficulties, model Verification, Validation and Uncertainty Quantification (UQ) have become indispensable in assessing predictive credibility. Here, model validation is defined as the process of authenticating the agreement between physical processes (typically obtained through experimentation) and the associated mathematical models. UQ is defined as ‘the process of quantifying and managing uncertainties associated with model calculations of true, physical quantities of interest, with the goals of accounting for all sources of uncertainty.’

Validation and UQ are critical for MFIX simulations because of the intricate interactions between the different physics in modeling a complex phenomenon (e.g., carbon capture, gasification). Errors and uncertainties first arise in formulating a set of governing equations for the physical process, including uncertainties in some model parameters in individual equations, uncertainties in the form of the equations (e.g., large eddy versus Reynold-averaged equations), and uncertainties in the coupling terms between different governing equations. More errors and uncertainties (e.g., discretization, algorithmic errors) are introduced in translating these equations into computer codes. These computer models are then validated with a set of experimental data. However, in practice, experimental data are scarce and they often involve noisy measurements. In addition, the computer models may need to be tuned to agree better with experiments. Finally, the validated computer models may be used to predict performance of a ‘virtual’ system that is an extrapolation from current systems where experimental data have been collected.

All these errors and uncertainties will undoubtedly call the predictive credibility of these models into question. As such, the overall process in establishing predictive credibility can become an insurmountable task. The project proposed to develop a unified framework to enable users with an average level of statistics knowledge to assess the credibility of MFIX-DEM simulation results by employing various UQ analysis methods through a tightly integrated UQ engine (i.e., PSUADE [24]) and Graphical User Interface (i.e., MFIX-GUI [1]). This effort aimed to yield a completely new and unique capability desirable by both industrial users and other institutional researchers. The proposed integrated uncertainty quantification framework relied on a graphical user interface (GUI) that interfaces with MFIX-DEM and provides a number of uncertainty quantification related tasks as depicted in Figure 56.

For Phase 1 of the project, a proof-of-concept prototype was targeted for development as part of the proposed framework to towards the goal of achieving an integrated, intuitive, graphical user interface based framework to facilitate ease of use and lower the barrier for adopting uncertainty quantification analysis workflows for modeling & simulation as shown

in Figure 56.

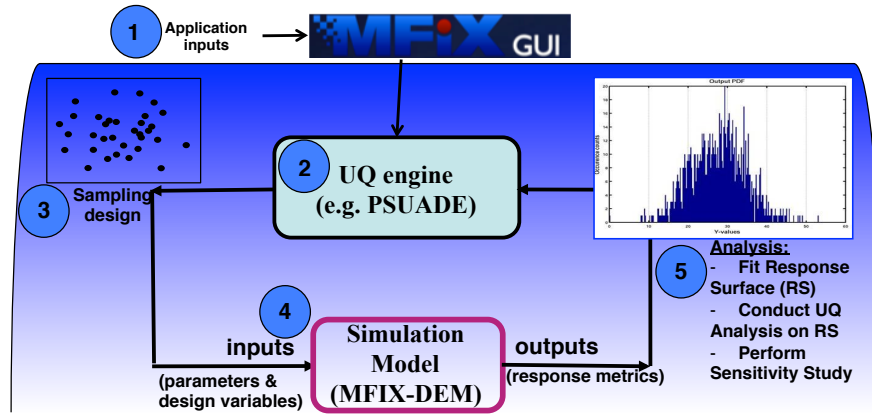


Figure 56. Depiction of an overview of the UQ capabilities and workflow embedded behind the GUI for MFIX-DEM (adapted from Gel et al.[14])

Figure 57 shows the proposed UQ workflow in a more detailed way and highlighted features that have been implemented for proof-of-concept prototype demonstration for Phase 1. The current proof-of-concept version is basically built upon an initial implementation using MFIX-GUI at the time of development (v17.1) and then utilizing the Nodeworks library which was originally developed by MFIX GUI team at NETL as part of the MFIX-GUI but then spun off as separate library.

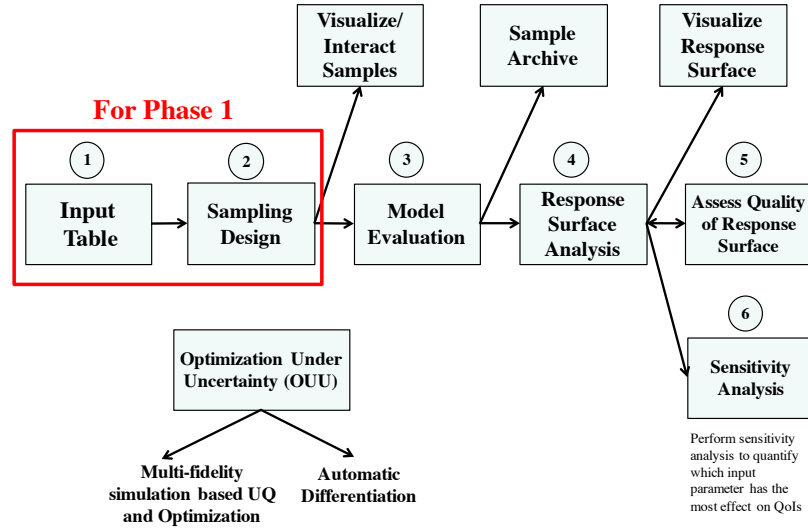


Figure 57. Illustration of the workflow proposed for UQ framework and proof-of-concept prototype implementation for Phase 1.

The fundamental idea of the proposed UQ framework was based on question driven and guided UQ analysis for the user who don't have the expertise in utilizing UQ analysis tools manually. Figures 58 to 62 show the operation of the current proof-of-concept version. As shown in Figure 58, once the user selects the UQ icon in the top bar, the user is posed with several questions that the answer is being sought as part of the UQ analysis. Although the questions have been compiled in layman terms, the appropriate UQ terminology for the analysis type is displayed in parenthesis to let the user become aware of the proper terminology used. For Phase 1 proof-of-concept demonstration only first question has been implemented, which is also shown in the rest of the figures. Once the user selects the question, a workflow consisting of multiple nodes that are relevant to the targeted function are automatically populated in the workbench as shown in Figure 59. Next, the user is asked to complete few entries for the nodes such as the variables to be considered and their range in the Variables and Correlations node, the total number of samples and sampling method for Sampling Design node as shown on Figure 60. After providing the necessary input for the UQ workflow for the selected question, the user triggers the analysis by clicking the play button. This initiates the process of compiling the input necessary for running PSUADE UQ toolkit from the GUI and interact with PSUADE behind the scenes by eliminating or at least minimizing the user's manual interaction with PSUADE. For the specific example, PSUADE generates a sample set based on the user selection (Figure 61) and then creates the input files to enable the user to manually launch the MFIX-DEM simulations. The current proof-of-concept was demonstrated for a simple case of hopper discharge problem with two input parameters corresponding to particle to particle and particle to wall coefficient of friction parameters and the user has asked for 12 samples as shown in Figure 62.

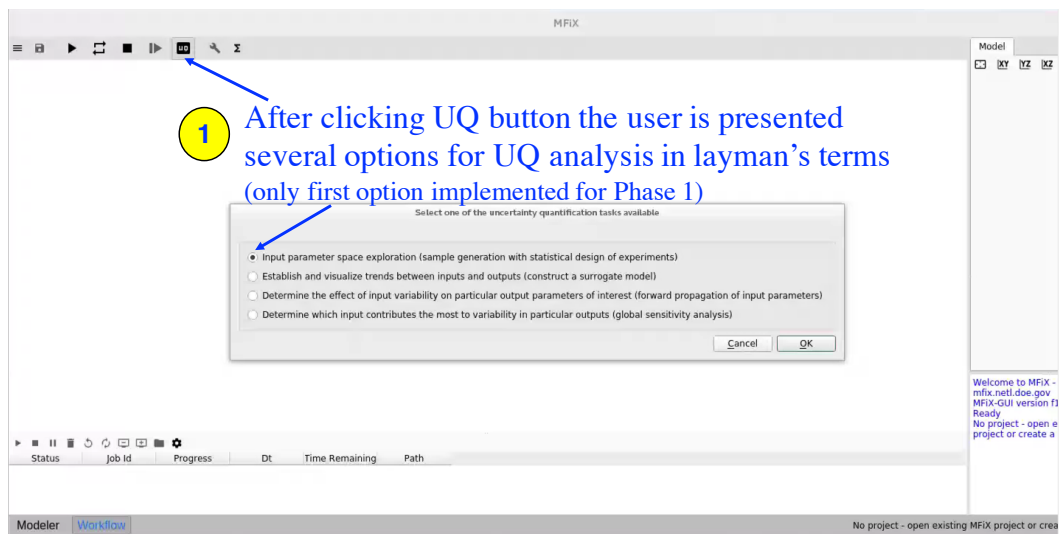


Figure 58. Illustration of the operation of the proof-of-concept prototype implemented for Phase 1: Step 1 user selects the UQ question that the answer is being sought

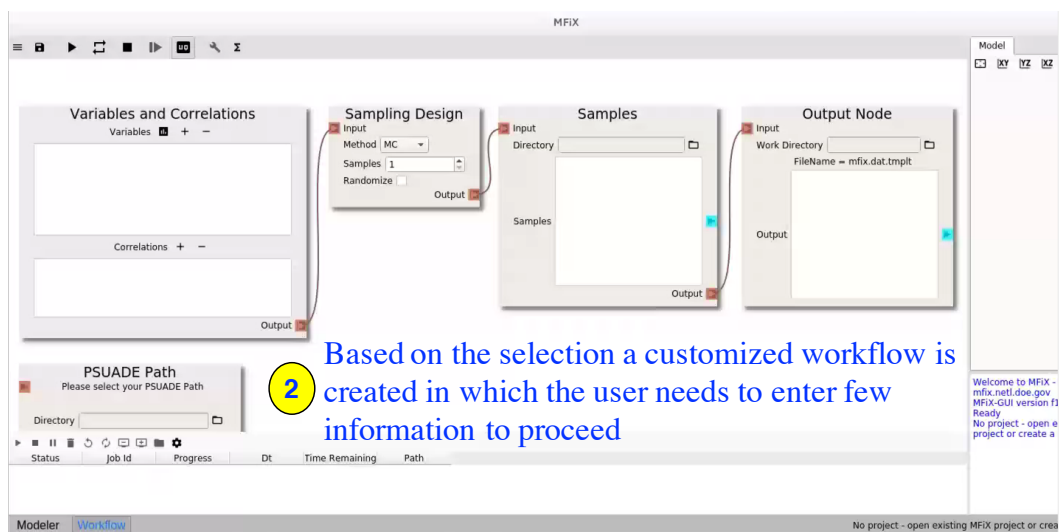


Figure 59. Step 2: UQ GUI sets up the nodes necessary to complete the UQ analysis automatically within Nodeworks framework based on the UQ question selected

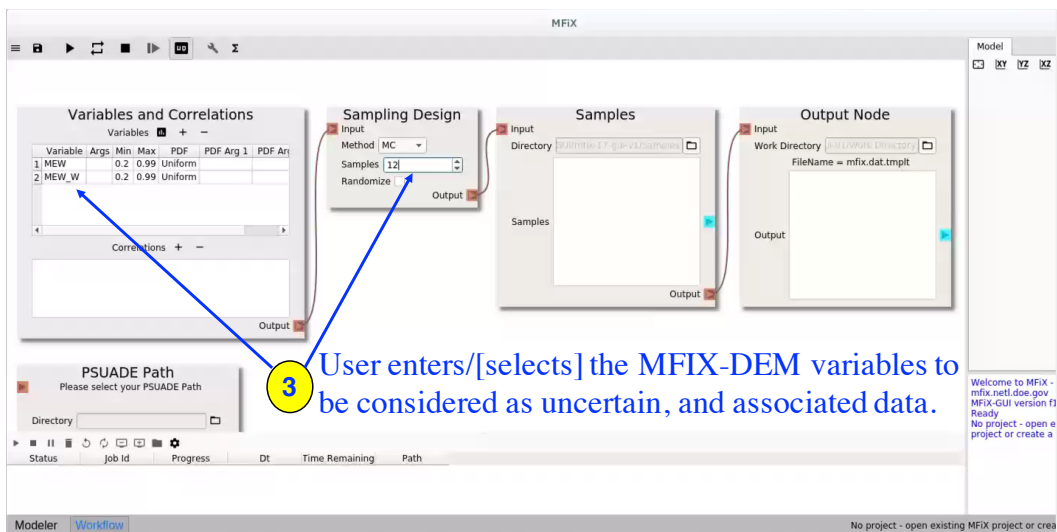


Figure 60. Step 3: User enters few required information such as the variables to be considered as uncertain, the lower and upper bounds, number of sampling simulations that can be afforded, etc.

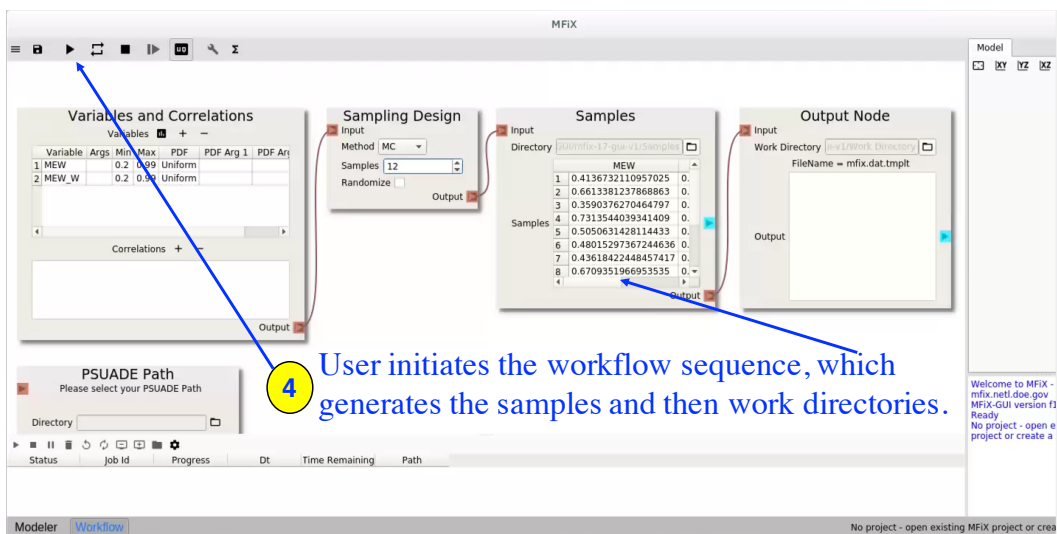


Figure 61. Step 4: User triggers the workflow by clicking the play button to run UQ GUI workflow, which in return interacts with PSUADE UQ toolbox behind the scenes.

```

[aike@ayvalik WorkDirectory]$ pwd
/home/aike/UQ_GUI/mfix-17-gui-v1/WorkDirectory
[aike@ayvalik WorkDirectory]$ ls -alrt
total 136
-rwxrwxr-x. 1 aike aike 7355 Nov 21 15:38 mfix_setupWorkdirs.py~
-rw-rw-r--. 1 aike aike 6009 Nov 21 15:38 mfix.dat.Tmplt
drwxrwxr-x. 13 aike aike 4096 Nov 27 23:22 ..
-rwxrwxr-x. 1 aike aike 7355 Nov 28 00:13 mfix_setupWorkdirs.py.bak
-rw-rw-r--. 1 aike aike 12407 Nov 28 00:14 psData_n5_i2_o1.bak
-rw-rw-r--. 1 aike aike 12727 Nov 28 00:14 psData_n5_i2_o1
-rwxrwxr-x. 1 aike aike 7355 Nov 28 00:14 mfix_setupWorkdirs.py
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.1
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.2
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.3
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.4
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.5
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.6
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.7
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.8
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.9
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.10
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.11
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 R.12
-rw-rw-r--. 1 aike aike 12727 Nov 28 00:14 psuadeData
drwxrwxr-x. 14 aike aike 4096 Nov 28 00:14 .
[aike@ayvalik WorkDirectory]$

[aike@ayvalik WorkDirectory]$
[aike@ayvalik WorkDirectory]$ cd R.1
[aike@ayvalik R.1]$ ls -alrt
total 32
-rw-rw-r--. 1 aike aike 6009 Nov 28 00:14 mfix.dat.Tmplt
-rw-rw-r--. 1 aike aike 44 Nov 28 00:14 psuadeApps_ct.in.1
-rw-rw-r--. 1 aike aike 70 Nov 28 00:14 psuadeApps_ct.out.1
-rw-rw-r--. 1 aike aike 6029 Nov 28 00:14 mfix.dat
drwxrwxr-x. 2 aike aike 4096 Nov 28 00:14 .
drwxrwxr-x. 14 aike aike 4096 Nov 28 00:14 ..
[aike@ayvalik R.1]$ diff mfix.dat ../mfix.dat.Tmplt
99,100c99,100
< MEW = 0.930994554051          ! particle-particle
< MEW_W = 0.510035532275       ! particle-wall
---
> MEW = PS_R1                  ! particle-particle
> MEW_W = PS_R2                ! particle-wall
[aike@ayvalik R.1]$

```

Figure 62. Step 5: Work directories to run independent MFIX-DEM job for this specific case is generated by creating 12 sampling runs where coefficient of friction for particle to particle and particle to wall are varied using Monte Carlo simulation.

A progress update presentation was provided to the MFIX GUI team at NETL on November 28th, 2017. The rest of the UQ workflow wouldn't be implemented as the demonstration of the proof-of-concept concludes the requirements for Phase 1. The proof-of-concept demonstration was later extended into Nodeworks as the workflow features originally employed within MFIX-GUI was abstracted and offered as standalone package in addition to MFIX-GUI. Nodeworks is a new workflow construction based graphical programming language being developed at NETL. The extension was carried out by incorporating the UQ wizard infrastructure and UQ nodes developed in the current project into Nodeworks. This environment offers a unique graphical user interface platform for the users and the goal is to lower the barrier for performing UQ analysis and capabilities for any type of modeling application by hiding complexity and interface with PSUADE within Nodeworks. Preliminary overview of this extension is provided in Figures 63 to 65.

Figure 63 shows the new UQ wizard concept which is called directly from the top left icons and poses several UQ analysis tasks for the user to choose (similar to the feature presented in Figure 58). Once the user decides and selects the task, the workflow area is populated with several nodes that are relevant to the UQ analysis task to be achieved as shown in Figure 64. Although the scope of Phase 1 proof-of-concept demonstration aimed very limited few tasks, as shown in Figure 65 many additional tasks such as Morris one at a time screening analysis, response surface construction were also included before the end of project. All of these new features were made with the support of lead Nodeworks developer and stored in a separate branch of Nodeworks git repository at NETL, which is automatically accessible to NETL participants.

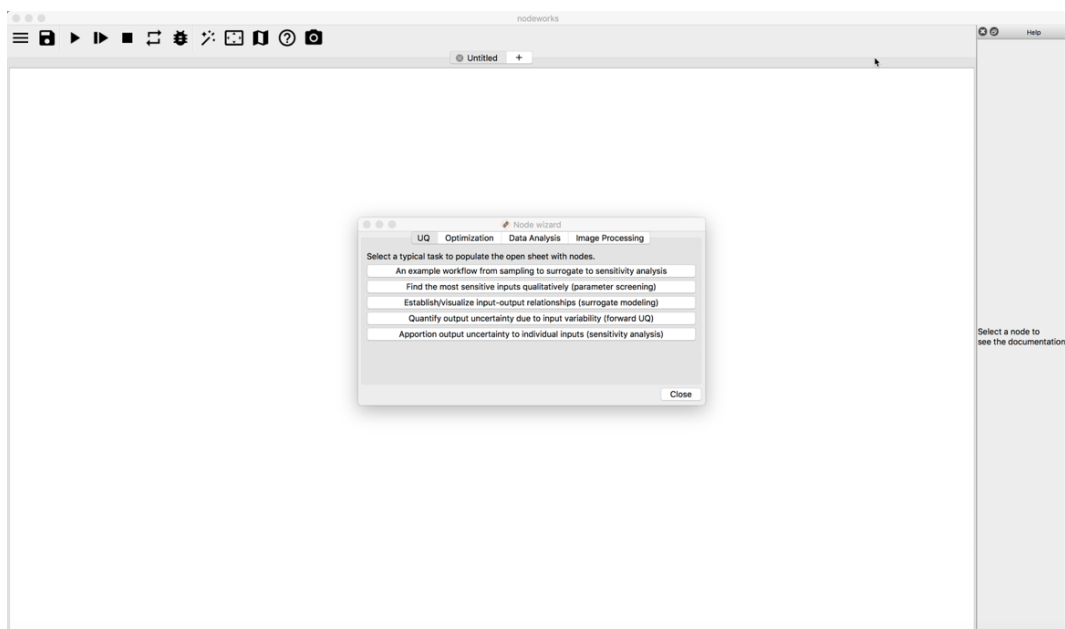


Figure 63. Illustration of the UQ wizard within Nodeworks with PSUADE integration.

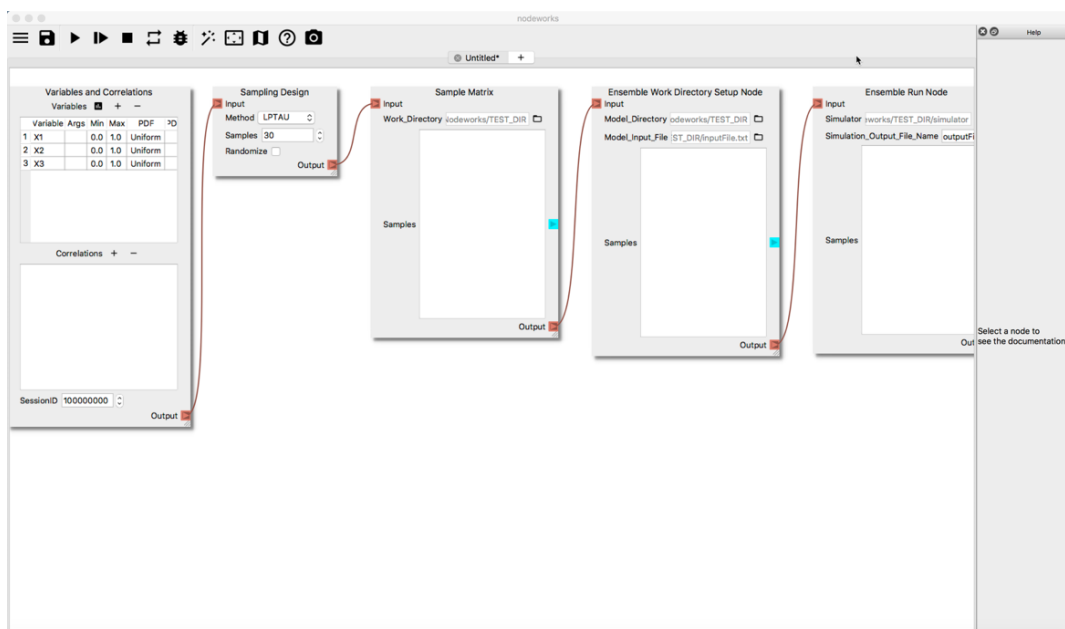


Figure 64. Illustration of the nodes populated within the workspace based on the user's selection from UQ wizard within Nodeworks with PSUADE integration.

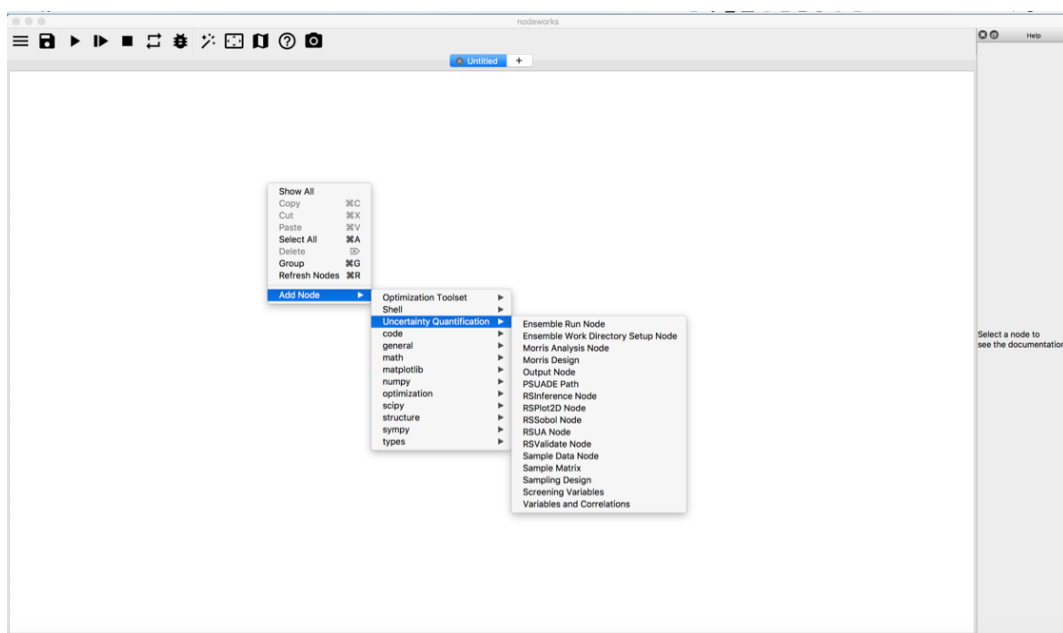


Figure 65. Illustration of the new UQ nodes added by porting from the original proof-of-concept demonstration within MFI-GUI into Nodeworks with PSUADE integration.

Task 5 - Perform Industry Outreach and Development of Industry Cases

An essential component of this project was the direct collaboration with industry from day one in order to lower the barrier for the use of MFI-X-DEM Phi by the industry for trouble shooting or scale-up analysis. In Phase 1, we have established active industrial collaborations with ExxonMobil, which is a world-leading oil company and Procter & Gamble, which is a major global consumer goods company at the beginning of the project and continued the collaboration through webex meetings and site visits including presentations on our project. Additionally, we were successful in initiating a new industrial collaboration with BASF Corporation during Phase 1. As part of these collaborations, industrial-relevant small-scale representative problems have been identified (i.e., hopper bin discharge and rotary drum with heat transfer) and extensively used as the prototype problems for initial testing, verification and validation of the implemented capability improvements, including handling particle polydispersity and validating heat transfer models. Valuable feedback on the design of lab-scale experiments at ASU for validation MFI-X-DEM for the benchmark problems and UQ design documents have also been acquired. Additional capability improvements in MFI-X-DEM crucial to industrial applications have been identified through a survey conducted among our industrial collaborators. A summary of findings from the survey results were shared with our NETL colleagues during progress update meetings. As part of our interactions with the industrial collaboration several site visits were organized, which gave us the opportunity to make general overview presentations to wider audience at each site to introduce MFI-X suite of solvers in addition to the technical work scope related presentations.

In the last several quarters of the project, another interesting problem was presented by our ExxonMobil collaborator, which was based on an existing issue and we proposed to explore with a proof-of-concept demonstration of modeling and simulation tools to address the problem. Details of the problem and our findings can be found under “Particle Height Classification Problem” in Subtask 3.1. The findings from this effort were collaboratively compiled in a manuscript titled “Classification of particle height in a hopper bin from limited discharge data using convolutional neural network models” [7] with our industrial collaborator as one of the co-authors. The manuscript which was submitted to Powder Technology journal, is under peer review at the time of writing this report.

Conclusions & Future Work

Recommendations

The project aimed to address the two major challenges that hinder the effective use and adoption of multiphase computational fluid dynamics by industry in a cohesive and technically-sound manner. Hence, all seven objectives given in the Subtopic 4B of the FOA (as shown in Figure 1) have been addressed through our project tasks since the start of Phase 1, as we believed that only an integrated and comprehensive action plan can offer a real developmental breakthrough for these non-trivial, interdependent challenges.

The results and outcomes of the Phase 1 activities are summarized with future work recommendations as follows.

A new feature for efficiently handling particle size polydispersity has been implemented in MFIX-DEM and has been systematically verified and validated via hopper bin discharge simulations. The improved MFIX-DEM codes have been utilized to generate training data for a novel particle height classification problem using CNN models, in collaboration with ExxonMobil. The next step for physical modeling capability enhancement is to implement non-spherical particle models (e.g., the family of superellipsoids). All of the industrial collaborators strongly urged the computationally efficient non-spherical particle model implementation in MFIX, which was the primary research task proposed for Phase 2 with a detailed plan based on analytical shape representations using superellipsoid family. The proposed approach was based on the prior extensive experience for modeling non-spherical (frictionless) granular systems using a variety of particle shape models [16, 17, 25, 18, 19, 20, 22].

As part of physical modeling enhancements, heat transfer models were reviewed and validated. In particular, for Phase 1 conduction heat transfer model was extensively validated with literature data, and the effects of operating parameters were studied with simulations and experiments. While rotation rate was determined to have no significant effect, particle size distribution and rolling friction had some effects based on the limited set of study performed. Hence, it is recommended to have a rolling friction model to be implemented and validated. Also it is recommended to perform extensive validation study for remaining heat transfer models, i.e., convection and radiation.

Several observations and recommendations based on industrial collaborator interactions:

- MFIX draw a lot of attention due to the unified framework for offering both CFD and DEM as opposed to externally coupling of two codes like OpenFOAM and LIGGGHTS, which most of the industrial collaborators had to use and complain about number of inefficiencies and problems. In spite of the unique advantage, it was observed that

MFIX was not adequately known by the industrial users. In many occasions they have either heard about it but didn't fully explore due to difficulties in installation or steep learning curve when someone attempted to try it in the past. The current simplified anaconda based installation method has lowered the barrier dramatically.

- The graphical user interface was another major contributor to get the attention of all these industrial collaborators. Hence, Mr. Justin Weber's willingness to help us demonstrate the earlier versions of it at the beginning of project was significant support.
- During recent site visit an engineer at P&G was so enthusiastic about his experience in being able to construct a two-phase flow model in couple hours and see meaningful results during his Christmas break, he did advertising for us to his managers and peers.
- More industrially relevant demonstration cases, which are simple enough to avoid intellectual property related complexities but easily extendable has substantial effect in getting the engineers of industrial collaborators to explore and test MFIX suite and report to their managers for use in their day to day operation.

Publications from the project (incl. under review or working papers):

- S. Chen, M. Adepu, H. Emady, Y. Jiao, A. Gel, "Enhancing the physical modeling capability of open-source MFIX-DEM software for handling particle size polydispersity: Implementation and validation", Powder Technology, 2017, 317:117-125, (<https://doi.org/10.1016/j.powtec.2017.04.055>)
- S. Chen, L. A. Baumes, A. Gel, M. Adepu, H. Emady, Y. Jiao, "Classification of particle height in a hopper bin from limited discharge data using convolutional neural network models", Powder Technology (Under review with id POWTEC-D-18-00754)
- M. Adepu, S. Chen, Y. Jiao, H. Emady, A. Gel, "Verification and Validation of the Conduction Heat Transfer Model in the Open-Source MFIX-DEM Code using a Rotary Drum" (working paper)

Conference Presentations & Poster Sessions:

- Y. Jiao, A. Gel, H. Emady, S. Chen, M. Adepu, C. Tong, O. Mor, J. Hu, N. Ellingwood, "MFIX-DEM Phi: Performance and Capability Improvements Towards Industrial Grade Open-source DEM Framework with Integrated Uncertainty Quantification", 2016 NETL Multiphase Flow Science Workshop, August 9, 2016, Morgantown WV
- M. Adepu, S. Chen, Y. Jiao, A. Gel, and H. Emady. "Quantitative Validation and Analysis of Heat Transfer in a Rotary Drum Using Experiments and Simulations". American Institute of Chemical Engineers (AIChE) Annual Conference Oral Presentation, 2017. URL <https://aiche.confex.com/aiche/2017/meetingapp.cgi/Paper/501016>.

- M. Adepu, S. Chen, Y. Jiao, A. Gel, and H. Emady. “Multilayer Granular Segregation in Discharging Cylindrical Hoppers“. American Institute of Chemical Engineers (AIChE) Annual Conference Oral Presentation, 2017. URL <https://aiche.confex.com/aiche/2017/meetingapp.cgi/Paper/501041>.

Presentations at the annual portfolio meeting organized by NETL are not included in the above list.

Products/Tools from the project:

- Polydispersity implemented version of MFIX-DEM, which was based on 2015-1 release of MFIX initially then first upgraded to 2016-1 and more recently 2017 release although full repeat of V&V tests couldn't be performed due to time constraints. The source code tree for 2017 release update has been provided to Dr. J. Dietiker for his review and incorporation with the most recent version of MFIX. Initial implementation was beta tested by NETL technical team (i.e., Dr. J. Dietiker, Dr. T. Li and Dr. Y. Xu in December 2016). More recently a senior staff member at LLNL is evaluating the new version for use in two separate HPC for Advanced Manufacturing projects at Lawrence Livermore National Laboratory (LLNL).
- Some of the capabilities with the polydispersity feature developed are being incorporated into MFIX-Exa project supported by Department of Energy's Exascale Computing Project (ECP).
- Uncertainty Quantification GUI prototype developed as part of Phase 1 has been adopted into another project at NETL as part of FWP for current fiscal year to offer Nodeworks (graphical programming framework and toolbox) with advanced UQ capabilities through PSUADE integration.

Industry Outreach and Collaboration:

- After the project start, we conducted site visits to ExxonMobil Research in New Jersey and Procter & Gamble in Ohio and Newcastle, UK to give presentations both to introduce MFIX suite solvers and also our proposed project activities.
- Survey completed by industrial collaborators
- Initiated a new collaboration with BASF Chemicals in the course of the project, which they contributed to several activities such as industry survey, discussion of the benchmark problem results, and suggestions for other cases.
- Periodic webex meetings with industrial collaborators for updates and to seek their feedback
- Submitted a jointly co-authored manuscript with ExxonMobil Chemicals collaborator

Students trained through the funded research effort: Two graduate students in doctorate programs were trained and acquired unique skills such as using remote HPC systems at national supercomputing centers, experimentation, application of machine learning libraries for computational science problems:

- Dr. S. Chen - Ph.D. student in Materials Science (who graduated from ASU in May 2018)
- Ms. M. Adepu - Ph.D. student in Chemical Engineering

Also the following undergraduate student worked during summer internship at LLNL for UQ GUI development and acquired skills in python programming, uncertainty quantification workflows:

Mr. O. Mor - Barrett Honors senior in Computer Science (now in Ph.D. program at University of Illinois at Urbana-Champaign)

External High Performance Computing Resources Acquired Through Peer-Review based Competitive Proposal Process without Any Cost to DOE/NETL: Table 5 shows the external computational resources acquired in Phase 1 performance period through competitive peer reviewed proposal submission process at the U.S. Department of Energy’s NERSC and National Science Foundation’s (NSF) XSEDE program. The NSF provided estimated value of the awarded XSEDE program HPC resources for 2017 & 2018 were 9, 127 and 5, 305, respectively for an approximate total of 14.4K.

Table 5. External HPC Resources Acquired Throughout Phase 1

	DOE/NERSC	NSF/XSEDE* (SDSC)	NSF/XSEDE* (TACC)
Year 1 (2016)	219,000 MPP Hrs (Edison/Cori-1)	50,000 SU (Startup)	50,000 SU (Startup)
Year 2 (2017)	464,000 MPP Hrs (Edison/Cori-2)		250,000 SU (Stampede)
Year 3 (2018)	200,000 MPP Hrs (Edison/Cori-2)	115,000 SU (Comet)	3312 SU (Stampede2)

References

- [1] MFIX-GUI, url = <https://mfix.netl.doe.gov/mfix/mfix-gui-2/>.
- [2] TAU Performance System, 2016.
- [3] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.
- [4] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.
- [5] S. Benyahia, M. Syamlal, and T. J. O’Brien. Summary of MFIX equations 2012-1.
- [6] Bodhisattwa Chaudhuri, Fernando J. Muzzio, and M. Silvina Tomassone. Experimentally validated computations of heat transfer in granular materials in rotary calciners. *Powder Technology*, 198(1):6–15, 2010.
- [7] S. Chen, L. A. Baumes, A. Gel, M. Adepul, H. Emady, and Y. Jiao. *Powder Technology submitted under peer review with id # POWTEC-D-18-00754*.
- [8] Shaohua Chen, Manogna Adepul, Heather Emady, Yang Jiao, and Aytakin Gel. Enhancing the physical modeling capability of open-source MFIX-DEM software for handling particle size polydispersity: Implementation and validation. *Powder Technology*, 317:117–125, 2017.
- [9] François Chollet et al. Keras: Deep learning library for theano and tensorflow. *URL: <https://keras.io/k>*, 2015.
- [10] Intel Corporation. Intel® Advisor™ XE 2016, 2016.
- [11] Intel Corporation. Intel® VTune™ Amplifier 2016, 2016.
- [12] Heather N Emady, Kellie V Anderson, William G Borghard, Fernando J Muzzio, Benjamin J Glasser, and Alberto Cuitino. Prediction of conductive heating time scales of particles in a rotary drum. *Chemical Engineering Science*, 152:45–54, 2016.
- [13] R. Garg, J. Galvin, T. Li, and S. Pannala. Documentation of open-source MFIX-DEM software for gas-solids flows.

- [14] A. Gel, R. Garg, C. Tong, M. Shahn timer, and C. Guent her. Applying uncertainty quantification to multiphase flow computational fluid dynamics. *Powder Technology*, 242:27–39, 2013.
- [15] Kevin A. Huck, Wyatt Spear, Allen D. Malony, Sameer Shende, and Alan Morris. Parametric studies in eclipse with tau and perexplorer. In *European Conference on Parallel Processing*, pages 283–294. Springer, 2008.
- [16] Y Jiao, FH Stillinger, and S Torquato. Optimal packings of superdisks and the role of symmetry. *Physical review letters*, 100(24):245504, 2008.
- [17] Yang Jiao, Frank H Stillinger, and Salvatore Torquato. Optimal packings of superballs. *Physical Review E*, 79(4):041309, 2009.
- [18] Yang Jiao, Frank H Stillinger, and Salvatore Torquato. Distinctive features arising in maximally random jammed packings of superballs. *Physical Review E*, 81(4):041304, 2010.
- [19] Yang Jiao and Salvatore Torquato. Communication: A packing of truncated tetrahedra that nearly fills all of space and its melting properties, 2011.
- [20] Yang Jiao and Salvatore Torquato. Maximally random jammed packings of platonic solids: Hyperuniform long-range correlations and isostaticity. *Physical Review E*, 84(4):041309, 2011.
- [21] W. R. Ketterhagen, J. S. Curtis, C. R. Wassgren, A. Kong, P. J. Narayan, and B. C. Hancock. *Chemical Engineering Science*.
- [22] Lingyi Meng, Yang Jiao, and Shuixiang Li. Maximally dense random packings of spherocylinders. *Powder Technology*, 292:176–185, 2016.
- [23] J. Musser, J. Carney, J. Dietiker, R. Garg, M. Meredith, and J. Weber. MFI X Update. In *Presented as part of the NETL Workshop on Multiphase Flow Science*. National Energy Technology Laboratory, 2014.
- [24] C. Tong. PSUADE reference manual (Version 1.7). *Lawrence Livermore National Laboratory, Livermore, CA*, 2015.
- [25] Salvatore Torquato and Yang Jiao. Dense packings of the platonic and archimedean solids. *Nature*, 460(7257):876–879, 2009.

