NUMERICAL AND EXPERIMENTAL ANALYSES OF ROOM AND HIGH TEMPERATURE DENSE, GRANULAR FLOWS COUPLED TO FLOW PROPERTY MEASUREMENTS FOR SOLAR THERMAL ENERGY STORAGE

A Thesis
Presented to
The Academic Faculty

by

Justin D. Yarrington

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
G. W. Woodruff School of Mechanical Engineering

Georgia Institute of Technology
August 2020

COPYRIGHT © 2020 BY JUSTIN D. YARRINGTON
NUMERICAL AND EXPERIMENTAL ANALYSES OF ROOM AND HIGH TEMPERATURE DENSE, GRANULAR FLOWS COUPLED TO FLOW PROPERTY MEASUREMENTS FOR SOLAR THERMAL ENERGY STORAGE

Approved by:

Dr. Peter Loutzenhisier, Co-advisor (ME)
G. W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Devesh Ranjan, Co-advisor (ME)
G. W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Zhuomin Zhang (ME)
G. W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Date Approved: July 21, 2020
To my wife, Loren
ACKNOWLEDGEMENTS

I thank my advisors, Drs. Peter Loutzenhiser and Devesh Ranjan, for their guidance, support, and encouragement. I also thank Dr. Zhuomin Zhang for his participation in my committee, and for his consideration and advice while completing this work. I also thank past and present members of the Solar Fuels and Technologies Laboratory and the Shock Tube and Advanced Mixing Laboratory, especially Malavika Bagepalli, Chuyang Chen, Gokul Pathikonda, and Andrew Schrader. Their insights and contributions have proven vital and have made this work more fulfilling and enjoyable. I thank my parents, sibling, and in-laws for their continued encouragement, love, and support, especially since beginning my graduate studies. I also thank my two beautiful daughters, Makena and Emma, for the happiness and love they constantly convey. Finally, to my wife, Loren: your endless support, constant faith, and unconditional love, have been instrumental to realizing the completion of this work and for that I am eternally grateful.

This material is based upon work supported by the U.S. Department of Energy’s Office of Energy Efficiency and Renewable Energy (EERE) under Solar Energy Technologies Office (SETO) Agreement Number EE0008372. This research was also supported in part through research cyberinfrastructure resources and services provided by the Partnership for an Advanced Computing Environment (PACE) at the Georgia Institute of Technology, Atlanta, Georgia, USA.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS iv

LIST OF TABLES vii

LIST OF FIGURES ix

LIST OF SYMBOLS AND ABBREVIATIONS xii

SUMMARY xvi

CHAPTER 1. Introduction 1
  1.1 Motivation and Literature Review 1
  1.2 Objectives 4
  1.3 Thesis Overview 5

CHAPTER 2. Room Temperature Flow Characterization 7
  2.1 Introduction 7
  2.2 Experimental Property Measurement 7
    2.2.1 Particle Shape and Size Distributions 9
    2.2.2 Elastic Properties 10
    2.2.3 Coefficient of Restitution 12
  2.3 Inclined Plane Flow Experimentation and Modeling 15
    2.3.1 Granular Flow Experiments 15
    2.3.2 Model Contact Mechanics 17
    2.3.3 Model Development 19
  2.4 Results and Discussion 22
    2.4.1 Particle Shape and Size Distributions 22
    2.4.2 Elastic Properties 28
    2.4.3 Coefficient of Restitution 30
    2.4.4 Model Validation with Experimental Flow Results 30
  2.5 Conclusions 36

CHAPTER 3. High Temperature Flow Characterization 37
  3.1 Introduction 37
  3.2 Measured Flow Properties in the DEM 37
    3.2.1 Particle Contact Mechanics 38
    3.2.2 Experimental Measurements of Particle Properties 40
  3.3 Particle Flow Model Development 47
    3.3.1 Model Mechanical Properties and Neighbour Pairing 47
    3.3.2 Flow Geometry and Particle Insertion 49
  3.4 Results and Discussions 52
  3.5 Conclusions 61

CHAPTER 4. Research Contributions and Future Work 64
  4.1 Research Contributions 64
4.2 Future Work

APPENDIX A. Data Processing Algorithms
A.1 Particle Size and Shape Distributions
A.1.1 Optical Microscopy Image Analysis
A.1.2 Size and Shape Distribution Algorithm
A.2 Elastic Properties
A.3 Coefficient of Restitution
A.3.1 Image Processing and PTV Algorithm
A.3.2 Post-processing Algorithms
A.4 Room Temperature Model Post-processing
A.4.1 Sub-function: Pubfig
A.4.2 Sub-function: Fmt_pubfig
A.5 High Temperature Model Post-processing
A.5.1 Mass Out Algorithm
A.5.2 Flow Behaviour Contours

REFERENCES
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2.1</td>
<td>List of dimensions used in the experimental setup to validate discrete element method model.</td>
<td>16</td>
</tr>
<tr>
<td>Table 2.2</td>
<td>A summary of LIGGGHTS contact models used to define the physical interaction between particles and their surroundings.</td>
<td>19</td>
</tr>
<tr>
<td>Table 2.3</td>
<td>Modeling parameters and resulting average velocity of a sensitivity study on dense granular flows along an inclined plane</td>
<td>21</td>
</tr>
<tr>
<td>Table 2.4</td>
<td>Probability distribution function parameters representative of measured particle shape and size data for Carbobead CP 30/60.</td>
<td>25</td>
</tr>
<tr>
<td>Table 2.5</td>
<td>Statistical shape and size parameters for Carbobead CP particles for various US standard mesh sizes.</td>
<td>28</td>
</tr>
<tr>
<td>Table 2.6</td>
<td>Mean and standard deviation of elastic properties for Carbobead CP specimen hot-pressed to densities equivalent to Carbobead CP particles</td>
<td>28</td>
</tr>
<tr>
<td>Table 2.7</td>
<td>Fitted parameters for elastic properties Spriggs’ correlations of Carbobead CP</td>
<td>30</td>
</tr>
<tr>
<td>Table 2.8</td>
<td>Mean coefficient of restitution for 10 Carbobead CP particle-to-particle collisions and standard deviation for separate US standard mesh size ranges</td>
<td>30</td>
</tr>
<tr>
<td>Table 2.9</td>
<td>Mechanical properties used in the simulation of inclined flow for Carbobead CP 30/60 particulates</td>
<td>31</td>
</tr>
<tr>
<td>Table 3.1</td>
<td>The average, scaled high temperature elastic properties of Carbobead CP particles at temperatures of interest.</td>
<td>43</td>
</tr>
<tr>
<td>Table 3.2</td>
<td>Mean and standard deviation of the coefficient of restitution of Carbobead CP 30/60 particle-particle impacts with corresponding system temperatures.</td>
<td>46</td>
</tr>
<tr>
<td>Table 3.3</td>
<td>Mean and standard deviation of the coefficient of restitution of Carbobead CP 30/60 particle-alumina impacts with corresponding system temperatures.</td>
<td>47</td>
</tr>
<tr>
<td>Table 3.4</td>
<td>Intrinsic and particle-particle mechanical properties of Carbobead CP 30/60 particles used in the simulation of Carbobead CP particle flows along an inclined plane.</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.5</td>
<td>Intrinsic and particle-wall mechanical properties of rigid alumina board used in the simulation of Carbobead CP particle flows along an inclined plane.</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.6</td>
<td>List of dimensions used in the inclined flow geometry.</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.7</td>
<td>Parameters used for particle insertion regions and particle inlet conditions.</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.8</td>
<td>The particle size distribution of each particle insertion region.</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.9</td>
<td>Parameters determined for linear trendlines of the calculated mass out.</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table 3.10</td>
<td>The times when steady state was reached, and the percent difference between the inlet and outlet mass flow rates for each temperature.</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Figure 2.1</td>
<td>Intensity as a function of $2\theta$ angle from x-ray diffractometry measurements of Carbobead CP flour and hot-pressed plate sample sintered at 25 MPa and 1400 °C.</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Schematic of impulse excitation test of rectangular test specimen including representative vibratory response after impulse for (a) torsional impact and mounting and (b) flexural impact and mounting.</td>
<td>11</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Coefficient of restitution experimental setup with the (a) SolidWorks rendering of the drop tube setup and (b) particle tracking of Carbobead CP particle prior to and after impact with hot-pressed Carbobead CP impact plane.</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Images of particle of the focal plane for an (a) in-focus particle and (b) out-of-focus particle, respectively.</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Room temperature experimental and modeling setup and for granular flow on an inclined plane with important features and dimensions shown for (a) inclined plane and hopper used in experiment and (b) modeling domain, and (c) hopper.</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.6</td>
<td>The flow geometry used in the sensitivity study of the mechanical properties. The average velocity magnitude within the region of interest was used as the primary output variable.</td>
<td>20</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>Optical microscopy image of Carbobead CP 30/60 particles from an optical microscope (Leica, USA).</td>
<td>23</td>
</tr>
<tr>
<td>Figure 2.8</td>
<td>Particle size characterization of Carbobead CP 30/60 particles from histograms with fitted lognormal probability distribution functions for (a) effective diameter, (b) major diameter, and (c) minor diameter.</td>
<td>26</td>
</tr>
<tr>
<td>Figure 2.9</td>
<td>Particle shape characterization of Carbobead CP 30/60 particles from histograms with fitted probability distribution functions for (a) circularity (b) roundness, and (c) aspect ratio</td>
<td>27</td>
</tr>
<tr>
<td>Figure 2.10</td>
<td>Elastic properties of Carbobead CP with Spriggs’ fit to (a) elastic modulus, (b) shear modulus, and (c) Poisson’s ratio as a function of material porosity</td>
<td>29</td>
</tr>
</tbody>
</table>
Figure 2.11  Images of PIV analysis on granular flow on an inclined plane with surface velocity vectors on a 100 × 8 mm² region of interest for different times after particle release including (a) 2.38 s, (b) 2.75 s, and (c) 5 s.

Figure 2.12  Free-surface velocities of particles along a 100 × 8 mm² region of interest on an inclined slope as a function of distance from the hopper along flow direction obtained using CDT and EPSD2 models from LIGGGHTS and measured velocities using PIV for different times after particle release including (a) 2.38 s, (b) 2.75 s, and (c) 5 s.

Figure 3.1  A spring-dashpot model describing particle interactions in the normal and tangential directions.

Figure 3.2  Measured elastic properties of Carbobead CP as a function of temperature with (a) the elastic and shear modulus, and (b) Poisson’s ratio.

Figure 3.3  Coefficient of restitution high temperature experimental setup with a schematic of (a) the alumina enclosure and (b) the particle drop and impact plane mechanisms.

Figure 3.4  Measured coefficient of restitution for Carbobead CP 30/60 particle-particle impacts as a function of temperature, where ‘+’ indicates outliers.

Figure 3.5  The geometry in a numerical analysis of room and high temperature particle flows along an inclined plane.

Figure 3.6  The insertion regions used to define inlet conditions for optimized DEM models of room and high temperature particle flows along an inclined plane.

Figure 3.7  Mass leaving the inclined plane flow geometry as a function of time for all temperatures.

Figure 3.8  Velocity magnitude fields of particle flows along an inclined plane at steady state with resolved polydisperse spheres for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.

Figure 3.9  Mass flux contours of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.
Figure 3.10  Volume fraction contours of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.

Figure 3.11  Average velocity magnitude contours of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.

Figure 3.12  Non-dimensional particle bed thickness of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.
## LIST OF SYMBOLS AND ABBREVIATIONS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>Length of particle insertion region</td>
</tr>
<tr>
<td>$A$</td>
<td>Projected area</td>
</tr>
<tr>
<td>$A_0$</td>
<td>Projected area prior to thermal expansion</td>
</tr>
<tr>
<td>$AR$</td>
<td>Aspect ratio</td>
</tr>
<tr>
<td>$b_e$</td>
<td>Material constant for elastic modulus</td>
</tr>
<tr>
<td>$b_g$</td>
<td>Material constant for shear modulus</td>
</tr>
<tr>
<td>$b_{in}$</td>
<td>Hopper inner breadth</td>
</tr>
<tr>
<td>$b_{out}$</td>
<td>Hopper outer breadth</td>
</tr>
<tr>
<td>$C$</td>
<td>Circularity</td>
</tr>
<tr>
<td>$C_{min}$</td>
<td>Minimum coordinates of a rectangular insertion region</td>
</tr>
<tr>
<td>$C_{max}$</td>
<td>Maximum coordinates of a rectangular insertion region</td>
</tr>
<tr>
<td>$d$</td>
<td>Diameter</td>
</tr>
<tr>
<td>$d_{eff}$</td>
<td>Effective particle diameter</td>
</tr>
<tr>
<td>$d_{major}$</td>
<td>Feret major diameter of particle</td>
</tr>
<tr>
<td>$d_{minor}$</td>
<td>Feret minor diameter of particle</td>
</tr>
<tr>
<td>$d_p$</td>
<td>Particle diameter</td>
</tr>
<tr>
<td>$E$</td>
<td>Elastic modulus</td>
</tr>
<tr>
<td>$E_0$</td>
<td>Elastic modulus of base material</td>
</tr>
<tr>
<td>$E_e$</td>
<td>Effective elastic modulus</td>
</tr>
<tr>
<td>$F$</td>
<td>Force acting on modeled particles</td>
</tr>
<tr>
<td>$F_{ij}$</td>
<td>View factor</td>
</tr>
<tr>
<td>$F_n$</td>
<td>Normal force</td>
</tr>
<tr>
<td>$F_f$</td>
<td>Frictional force</td>
</tr>
<tr>
<td>$F_t$</td>
<td>Tangential force</td>
</tr>
<tr>
<td>$G$</td>
<td>Shear modulus</td>
</tr>
<tr>
<td>$G_0$</td>
<td>Shear modulus of base material</td>
</tr>
<tr>
<td>$G_e$</td>
<td>Effective shear modulus</td>
</tr>
<tr>
<td>$h$</td>
<td>Height of a 2D inlet plane</td>
</tr>
<tr>
<td>$\bar{h}$</td>
<td>Non-dimensional bed height</td>
</tr>
<tr>
<td>$h_{bot}$</td>
<td>Hopper bottom region height</td>
</tr>
<tr>
<td>$h_{gap}$</td>
<td>Height of hopper-plane clearance gap</td>
</tr>
<tr>
<td>$h_{in}$</td>
<td>Hopper inner height</td>
</tr>
<tr>
<td>$h_{max}$</td>
<td>Overall maximum bed height within a model</td>
</tr>
<tr>
<td>$h_{out}$</td>
<td>Hopper overall outer height</td>
</tr>
<tr>
<td>$h_r$</td>
<td>Maximum particle bed height within a region</td>
</tr>
<tr>
<td>$k$</td>
<td>Elastic constant</td>
</tr>
<tr>
<td>$l$</td>
<td>Perpendicular distance between COR heater and a particle</td>
</tr>
<tr>
<td>$L_1$</td>
<td>Length between inclined plane edge to bottom of flow insertion region</td>
</tr>
<tr>
<td>$L_2$</td>
<td>Length between flow insertion region and bottom inclined plane edge</td>
</tr>
<tr>
<td>$L_{flow}$</td>
<td>Inclined plane flow length</td>
</tr>
<tr>
<td>$L_{plane,total}$</td>
<td>Inclined plane total length</td>
</tr>
<tr>
<td>$L_{stb}$</td>
<td>Flow stabilization length</td>
</tr>
</tbody>
</table>
$L_p$  
Length along plane region used for particle image velocimetry

$L_{PIV}$  
Camera coverage length for particle image velocimetry

$m$  
Mass

$m$  
Mass flow rate

$mc$  
Effective particle mass

$mc,t$  
Experimental mass out

$m_{23^\circ C,t}$  
Mass out of full-scale 23 °C model

$m_{23^\circ C}$  
Mass out of half-scale 23 °C model

$m_{200^\circ C}$  
Mass out of 200 °C model

$m_{400^\circ C}$  
Mass out of 400 °C model

$m_{600^\circ C}$  
Mass out of 600 °C model

$m_{800^\circ C}$  
Mass out of 800 °C model

$n$  
Normal unit vector

$\bar{Nu}_D$  
Average Nusselt number

$p$  
Porosity

$P$  
Perimeter

$P_0$  
Perimeter prior to thermal expansion

$Pr$  
Prandtl number

$R$  
Roundness

$Re$  
Equivalent radius of two interacting particles

$Re_D$  
Reynolds number based on particle diameter

$r$  
Particle radius

$r_h$  
Effective radius of COR heater

$s$  
Skin parameter

$S$  
Constant used in determining the view factor

$t$  
Time elapsed

$t_s$  
Time when steady state was reached

$T$  
Temperature

$T_{p,drop}$  
Particle temperature at release

$T_{impact}$  
Impact plane temperature

$T_{p,impact}$  
Particle temperature at impact

$U$  
Particle surface velocity

$V$  
Particle velocity

$V_{particle}$  
Modeled particle velocity

$V_{after}$  
Particle velocity after collision

$V_{before}$  
Particle velocity before collision

$\bar{V}_m$  
Average velocity magnitude

$w$  
Width of a 2D inlet plane

$W$  
Inclined plane width

$w_{DEM}$  
Width of modeling domain

$w_{in}$  
Hopper inner width

$w_{out}$  
Hopper outer width

$\bar{x}$  
Particle position vector
Greek Letters

$\alpha$  Rebound angle for coefficient of restitution
$\alpha_l$  Linear thermal expansion coefficient
$\beta$  Material constant used to determine interacting forces in LIGGGHTS
$\beta_0$  Mass intercept of linear trendlines
$\beta_1$  Slope of linear trendlines
$\gamma$  Shape parameter for Weibull distribution
$\gamma_{\text{damp}}$  Viscoelastic damping constant
$\delta_{\text{in}}$  Hopper outlet clearance
$\delta_{\text{out}}$  Hopper bottom clearance
$\delta_{\text{overlap}}$  Modeled particle overlap
$\varepsilon$  Coefficient of restitution
$\kappa$  Scale parameter for Weibull distribution
$\phi$  Standard deviation of random variable for lognormal probability distribution function
$\varphi$  Volume fraction
$\mu$  Coefficient of friction
$\mu_s$  Coefficient of static sliding friction
$\mu_r$  Coefficient of static rolling friction
$\nu$  Poisson’s ratio
$\nu_0$  Poisson’s ratio of base material
$2\theta$  Incident angle of x-ray in x-ray diffractometry
$\rho$  Density
$\rho_s$  Density of specimen
$\rho_t$  Theoretical density of base material
$\tau_r$  Rolling friction torque
$\theta_{\text{hopper}}$  Hopper angle
$\theta_{\text{plane}}$  Plane angle of inclination
$\Delta \theta_p$  Relative rotation between two interacting particles
$\theta_r$  Rolling friction angle
$\omega$  Mean of random variable for lognormal probability distribution function
$\omega_{r,s}$  Relative angular velocity between two interacting particles

Subscripts

e  Effective constant
eff  Effective property
i  Particle number/type $i$
j  Particle number/type $j$
\( n \) Normal direction
\( t \) Tangential direction
stb Stabilization

**Superscripts**
^ Unit vector
" Flux
\( \rightarrow \) Vector
— Mean value
\( \sim \) Normalization
° Degree

**Acronyms**
ANOVA Analysis of variance
CDT Constant directional torque
CP Carbobead CP
CSP Concentrated solar power
DEM Discrete element method
EPSD Elastic-plastic-spring-dashpot
EPsd2,3 Alternative elastic-plastic-spring-dashpot
HSP Carbobead HSP
ID50 Accucast ID50
LIGGGHTS LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator)
Improved for General Granular and Granular Heat Transfer Simulations
Pdf Probability distribution function
PIV Particle image velocimetry
PTV Particle tracking velocimetry
SPHRR Solar particle heating receivers/reactors
TES Thermal energy storage
XRD X-ray diffraction
SUMMARY

Solar thermal energy storage using sintered bauxite particles as a storage media is a useful tool for extending the operation and increasing operating temperatures of concentrated solar power systems (CSP). The flow behavior of sintered bauxite particles was characterized in this work to better inform the design of next generation CSP technologies.

Room temperature granular flows of sintered bauxite particles were examined along an inclined plane. Flow properties needed to drive numerical granular models were measured to improve model predictions for Carbobead CP particles. Particle shape and size distributions were determined by coupling optical microscopy to an in-house image processing algorithm. The impulse excitation technique was used to measure elastic and shear moduli, and compute Poisson’s ratio. The coefficient of restitution was measured by dropping particles on a surface and determining the kinetic energy before and after impact using high resolution particle tracking velocimetry. An inclined flow experiment was performed to characterize granular flows of Carbobead CP particles using particle image velocimetry. Numerical models of the experiment using the discrete element method were built with the measured flow properties and compared with experimental results. High temperature flow properties were measured to predict the high temperature flow behavior for Carbobead CP particles up to 800 °C. A numerical flow model at room temperature was extended to high temperature using the measured flow properties to determine the influence of temperature on the flow behavior.
CHAPTER 1. INTRODUCTION

1.1 Motivation and Literature Review

The mitigation of climate change and carbon emissions are some of the leading objectives in the pursuit to establish energy security. Renewable energy technologies have presented a promising pathway to realize these ambitions. Solar technologies in particular can realize these aims by reducing energy consumption and transportation emissions[1-3] while simultaneously achieving national goals for energy security[4]. However, the utilization and broader adoption of solar technologies is not without significant challenges. While sunlight incident on the earth seems abundant, solar irradiation is intermittent, dilute, and unequally distributed across the surface of the earth.

These challenges can be mitigated by developing concentrating solar power (CSP) technologies that utilize thermal energy storage (TES). When solar energy is focused to a central region several 100s or 1000s of times, the concentrated solar irradiation can be converted to useful thermal energy via absorption by a heat transfer medium. The absorbed thermal energy can then be used to generate electricity via a heat engine, such as an air-Brayton or supercritical CO₂ power cycle. High solar concentration ratios enable operation at higher working temperatures, leading to higher theoretical efficiencies, and Solar concentration ratios of more than 1000 suns are achievable during peak operating conditions at advanced CSP facilities (1 sun equals 1 kW/m²) [5, 6]. Theoretically, solar-to-work potential (exergy) efficiencies as high as 60% are achievable for a blackbody cavity receiver coupled to a Carnot heat cycle, by operating above 1000 °C for a concentration ratio of 1000 suns [7].
The major challenge associated with CSP technology (and any solar thermal technology) is to efficiently store the thermal energy at high temperature to enable on-demand production of electricity during periods when sunlight is unavailable. TES is an attractive pathway to store thermal energy due to its ability to extend the operating hours of CSP facilities, resulting in improved cost-competitiveness [8]. However, many obstacles such as heat transfer media costs, high pressure components, and temperature constraints must first be addressed. Molten salts are currently widely implemented as storage media, but are severely constrained by their operating temperatures, with freezing occurring at less than 200 °C, and thermal dissociation occurring greater than 600 °C [9]. The use of inert particles for TES has become an attractive alternative to circumvent some of the challenges associated with molten salts. Various particle types, particularly sintered bauxite, have been investigated because they are both chemically inert and thermally stable above 1000 °C [10-12].

Sintered bauxite particles have been investigated in solar particle heating receivers/reactors (SPHRR) for falling-particle receivers [9-11, 13-15] and centrifugal particle receivers [16, 17], and they are commercially available in different size ranges. Carbobead CP particles (known formerly as ACCUCAST ID) are composed of Al₂O₃, SiO₂, Fe₂O₃, and TiO₂ with a high solar absorptance greater than 90% [13]. Particle-based solar receivers used to enhance particle heat absorption are also relatively inexpensive, with a decrease in estimated marginal costs of as much as 37.5% compared to molten salts [18]. The high solar absorptance facilitates direct exposure to concentrated irradiation and high receiver efficiencies. The particles are also excellent high temperature TES media due to their large heat capacity. The direct utilization of these particles for TES eliminates
additional heat losses, irreversibilities, and costs associated with incorporating secondary storage medium.

Fundamental properties of refractory materials have been investigated for TES and heat transfer in CSP [19-22]. These properties inform the study of CSP technology using discrete element method (DEM) models, including receivers [23-25], moving bed heat exchangers [26, 27], and fluidized bed reactors [28]. Mechanical properties characterized at ambient conditions have provided useful information on the design of CSP systems [19, 20]. Therefore, understanding the behavior of granular flows at ambient temperature provides an important foundation for modeling granular flows into SPHRRs and transport throughout the system. This also provides the fundamental building blocks for understanding high-temperature flows that occur in various components in CSP systems. Changes in high temperature granular flow behavior have been observed experimentally in a solar thermochemical reactor, and significant changes in the velocity fields and agglomeration formation were observed for flows at elevated temperatures [29]. Therefore, understanding the impact of temperature on granular flow behavior is imperative for designing and optimizing the next generation of CSP solar particle heating receivers/reactors (SPHRR).

Empirical and computational studies have been performed to characterize bulk granular transport. Particle image velocimetry (PIV) was implemented in studying the flow distribution for different flow configurations of a variety of granular media, including inclined planes [30-33] and silos [34, 35]. Granular flow models for various applications, including particle-based CSP, have been investigated computationally using the DEM [23, 36-38]. LIGGGHTS [LAMMPS (Large-scale Atomic/Molecular Massively Parallel
Simulator) Improved for General Granular and Granular Heat Transfer Simulations] is an open-source DEM software that has been used to investigate granular flows in CSP applications [23, 24, 39, 40].

The particle shape and size, elastic properties, coefficient of restitution, and friction coefficients are the primary modeling inputs for DEM to capture particle-particle and particle-surface interactions [21, 23, 36, 41]. Fitted properties have been used to improve simulation run-time efficiencies [21, 23, 36, 41] in the absence of available flow properties. However, this results in a loss of physical meaning of the properties as different combinations of input flow properties produced similar modeled granular flow behavior [36, 42]. Future DEM models of granular flows for TES would benefit from measured flow properties of candidate TES particulate media.

1.2 Objectives

The cost-competitiveness of CSP has been greatly enhanced with newly developed TES techniques [8], and utilizing particulate TES media has demonstrated particular promise in developing next-generation CSP infrastructure. However, investigations of candidate TES media have indicated a significant gap in the knowledge of the flow behavior of candidate particulate media [29, 43]. Addressing this knowledge gap via the characterization of the flow behavior of particulate TES media will greatly assist in the design of next generation SPHRRs.

The aim of this work is to investigate the flow behavior of granular flows of particulate TES media at room and high temperatures. Flow properties of Carbobead CP particles at room and high temperature (sintered bauxite from Carbo Ceramics Inc.) were
measured\textsuperscript{1}. An inclined plane granular flow experiment was performed at room temperature, and DEM models of the experiment were produced using measured mechanical properties. Experimental free-surface velocities were measured with PIV and compared to the DEM model results. The validated room temperature DEM model was then extended to high temperatures with measured flow properties to determine the effect of temperature on the flow behavior. The high temperature models determined the steady state flow behavior of Carbobead CP particles at several temperatures up to 800 °C.

1.3 Thesis Overview

The thesis is organized into two major parts comprising the room temperature and high temperature flow behavior characterizations. In Chapter 2, experimental measurements of flow properties including particle shape and size distributions, elastic properties, and coefficient of restitution are presented for Carbobead CP sintered bauxite particles. The measured flow properties were used to develop a predictive DEM model of granular flow along an inclined plane. The model was experimentally validated via a comparison of the transient and steady-state surface velocity fields of the model and experiment.

In Chapter 3, high temperature flow property measurements of Carbobead CP particles are presented for temperatures up to 800 °C. Additionally, the room temperature flow model is optimized and extended to high temperature to determine the influence of temperature on the flow behavior. The velocity fields, average flow velocity, flow volume

\textsuperscript{1} Measurements were completed for particle shape and size; elastic moduli and Poisson’s ratio; and coefficient of restitution. Measurements of the coefficients of static sliding and static rolling friction were completed by a colleague and are not presented in this work.
fraction, flow bed thickness, and mass flux contours from flows at 23 °C, 200 °C, 400 °C, 600 °C, and 800 °C are compared.

In Chapter 4, the technical contributions of this work are reviewed. Additional opportunities for research and recommendations for future work are also presented.
CHAPTER 2. ROOM TEMPERATURE FLOW CHARACTERIZATION

2.1 Introduction

An investigation of room temperature granular flows of sintered bauxite proppants along an inclined plane for TES are presented. Room temperature flow properties of Carbobead CP particles (sintered bauxite from Carbo Ceramics Inc.) were measured for particle shape and size; elastic moduli and Poisson’s ratio; and coefficient of restitution. Several techniques to measure the room temperature properties were employed and include optical microscopy, impulse-excitation of hot-pressed particle specimen, and particle impact collision tests. A granular flow experiment was performed at room temperature and DEM numerical models of the experiment were produced using measured mechanical properties. Measurements of the coefficients of static sliding and static rolling friction, completed by a colleague, were also included in the DEM models [43]. Experimental free-surface velocities were measured with PIV and compared to DEM model results.

2.2 Experimental Property Measurement

Intrinsic properties (properties not dependent on shape or size) were measured using hot-pressed plates produced from Carbobead CP Flour (< 40 µm), and the extrinsic

---

2 Flow property measurements including elastic properties, particle shape and size distributions, coefficients of static friction, and the granular flow experiment were completed in collaboration with Malavika V. Bagepalli and Dr. Andrew Schrader. More information on the coefficients of static friction measurements are given in:

properties (properties dependent on shape and size) were measured for three different U.S standard mesh size ranges of Carbobead CP particles: (1) 50/140, (2) 40/100, and (3) 30/60, based on availability from the manufacturer. Carbobead CP flour was hot-pressed for 2 h in a graphite mold at temperatures between 1300 to 1500 °C and pressures between 20 to 25 MPa, resulting in different plate porosities. The plates were machined into different geometries to measure different properties.

X-Ray diffractometry (XRD, Malvern PANalytical Empyrean) was performed on both the Carbobead CP powder and hot-pressed Carbobead CP plates to ensure uniform compositions. The intensity as a function of 2θ angle from XRD measurements of Carbobead CP Flour and hot-pressed plate are shown in Figure 2.1. The intensity peaks for the hot-pressed plate were scaled and shifted up for comparison with the Carbobead CP Flour for 2θ values between 20 and 90°. A peak around 2θ = 45° was measured after hot-pressing, indicative of a slight deviation in structure possibly due to the introduction of an impurity. Nevertheless, the majority of the intensity peaks were indistinguishable with no detectable crystalline changes.
2.2.1 Particle Shape and Size Distributions

Optical microscopy was used to capture high resolution 2D images of Carbobead CP particles at ambient conditions to quantify the shape and size. The images were analyzed using an in-house image processing algorithm to measure both particle shape and size distributions. Particle size was characterized using a major diameter, minor diameter, and effective diameter. The effective diameter \( d_{\text{eff}} \) was calculated as:

\[
d_{\text{eff}} = 4 \frac{A}{P},
\]

where \( A \) is the projected area; and \( P \) is the perimeter of the particle in the 2D image. The \( d_{\text{eff}} \) represents an equivalent circle diameter. The particle shape was characterized using
circularity, roundness, and aspect ratio. Circularity and roundness have been used interchangeably [44, 45]. Circularity [44], roundness [46], and aspect ratio [45] were determined, respectively, as:

\[ C = 4\pi \frac{A}{P^2}, \]  

\[ R = \frac{4A}{\pi d_{\text{major}}^2}, \]  

\[ AR = \frac{d_{\text{major}}}{d_{\text{minor}}}, \]  

where \( d_{\text{major}} \) and \( d_{\text{minor}} \) are the major and minor diameters, respectively, which were determined as the maximum and minimum Feret diameters [47], respectively. \( C \) accounts for the overall circular quality of the particles using \( A \) and \( P \), and \( R \) accounts for the circular quality of particle with respect to maximum Feret diameter.

2.2.2 Elastic Properties

The elastic and shear moduli were measured with the hot-pressed Carbobead CP plates using the impulse excitation method, and Poisson’s ratio was calculated from these measurements.

A schematic of the impulse excitation setup is shown in Figure 2.2 for (a) the flexural and (b) the torsional impulse excitation testing of a rectangular test specimen with representative specimen vibratory responses. The dynamic harmonic frequency of a test specimen was recorded during a test using a microphone monitoring the vibrational
responses of the test specimen to a mechanical impulse. The natural resonant frequency of the test specimen is correlated to the geometry, mass, and material mechanical properties. Both elastic and shear moduli were measured based on test specimen support and impulse locations. Poisson’s ratio was determined as a function of the elastic and shear modulus [48], given as:

\[ \nu = \left( \frac{E}{2G} \right) - 1, \]  

(2.5)

where \( E \) is the elastic modulus and \( G \) is the shear modulus.

Figure 2.2 Schematic of impulse excitation test of rectangular test specimen including representative vibratory response after impulse for (a) torsional impact and mounting and (b) flexural impact and mounting.

Test specimens were produced for Carbobead CP materials by machining the hot-pressed plate into rectangular homogeneous specimens of 3×4×50 mm³ with different porosities to measure porosity dependent elastic properties. Elastic properties for a
specified material porosity were fit to the empirical Spriggs’ correlations [49, 50], given for $E$, $G$, and $\nu$, respectively, as:

\begin{align*}
E &= E_0 \exp(-b_e p), \\
G &= G_0 \exp(-b_g p), \\
\nu &= (\nu_0 + 1) \exp[-(b_e - b_g)p] - 1,
\end{align*}

(2.6) \quad (2.7) \quad (2.8)

where $E_0$, $G_0$, and $\nu_0$ are the elastic and shear moduli, and Poisson’s ratio of the theoretical density material, respectively; and $b_e$ and $b_g$ are material constants for elastic and shear moduli, respectively; and $p$ is the density-dependent specimen porosity, defined as:

\[ p = \left[ 1 - \frac{\rho_s}{\rho_t} \right], \]

(2.9)

where $\rho_s$ is the specimen density and $\rho_t$ is the theoretical density of the base material.

2.2.3 Coefficient of Restitution

The coefficient of restitution is given as:

\[ \epsilon = \frac{\vec{V}_{\text{after}} \cdot \hat{n}}{\vec{V}_{\text{before}} \cdot \hat{n}}, \]

(2.10)

where $\vec{V}_{\text{after}}$ and $\vec{V}_{\text{before}}$ are the particle velocity vectors after and before impact, respectively; and $\hat{n}$ is the normal unit vector to the impact surface. Particle-particle $\epsilon$ was measured for Carbobead CP 50/140, 40/100, and 30/60 particles by dropping them on a
hot-pressed Carbobead CP impact plane. The experimental setup is shown in Figure 2.3 with (a) SolidWorks rendering of the drop-tube setup and (b) particle tracking of Carbobead CP particle prior to and after impact with the impact plane. The drop tubes were positioned within a sealed chamber to prevent airflow from disrupting particles during drops. A high-speed camera (Photron SA3, 512 × 768 pixel²) was used to record particle motion at a rate of 2000 frames/s. Distortions of particles with motion perpendicular to the camera view were identified visually as they moved out of the focal plane. Particles leaving the focal plane were excluded in coefficient of restitution calculations. A distinction between a particle inside and outside of the focal plane is shown in Figure 2.4 for (a) an in-focus particle and (b) an out-of-focus particle, respectively. Particle tracking velocimetry (PTV) was used to determine $\bar{V}_{\text{after}} \cdot \hat{n}$ and $\bar{V}_{\text{before}} \cdot \hat{n}$ calculate $\varepsilon$. The rebound angle of particles relative to $\hat{n}$ after impact depended on particle shape, where non-spherical particles demonstrated high rebound angles [51]. Measurements with rebound angles within $\pm 10^\circ$ and within the focal plane were chosen to obtain intrinsic values of $\varepsilon$. 
Figure 2.3 Coefficient of restitution experimental setup with the (a) SolidWorks rendering of the drop tube setup and (b) particle tracking of Carbobead CP particle prior to and after impact with hot-pressed Carbobead CP impact plane.

Figure 2.4 Images of particle of the focal plane for an (a) in-focus particle and (b) out-of-focus particle, respectively.
2.3 **Inclined Plane Flow Experimentation and Modeling**

A room temperature granular flow experiment was completed, and a corresponding numerical model of the experiment was developed using the DEM with measured mechanical properties. The inclined flow geometry used for modeling and experimentation is discussed, and the contact mechanics implemented in the DEM simulation are presented.

2.3.1 *Granular Flow Experiments*

An inclined flow experiment was performed to examine granular flows of Carbobead CP 30/60. A schematic of the inclined flow experiment is shown in Figure 2.5 for (a) inclined plane and hopper used in experiment and (b) modeling domain, and (c) hopper, with dimensions listed in Table 2.1. The plane was fabricated from rigid alumina board coated with a high-purity alumina coating (Aremco PyroPaint 634-AL) to create a roughened, erosion resistant surface. A suitable plane inclination from the horizontal, $\theta_{\text{plane}}$, was used that enabled stable, non-accelerated flows. The hopper was fabricated using a resin-based 3D printer (Formlabs Form 2) and mounted parallel to plane surface.
Figure 2.5 Room temperature experimental and modeling setup and for granular flow on an inclined plane with important features and dimensions shown for (a) inclined plane and hopper used in experiment and (b) modeling domain, and (c) hopper.

Table 2.1 List of dimensions used in the experimental setup to validate discrete element method model.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_{in}$</td>
<td>50.8</td>
</tr>
<tr>
<td>$L_{PIV}$</td>
<td>100</td>
</tr>
<tr>
<td>$b_{out}$</td>
<td>$w_{out}$</td>
</tr>
<tr>
<td>$L_{PIV}$</td>
<td>$w_{DEM}$</td>
</tr>
<tr>
<td>$h_{bot}$</td>
<td>9.7</td>
</tr>
<tr>
<td>$w_{flow}$</td>
<td>80</td>
</tr>
<tr>
<td>$h_{in}$</td>
<td>34.3</td>
</tr>
<tr>
<td>$w_{flow}$</td>
<td>63.5</td>
</tr>
<tr>
<td>$h_{out}$</td>
<td>76.2</td>
</tr>
<tr>
<td>$\delta_{in}$</td>
<td>2.4</td>
</tr>
<tr>
<td>$L_{flow}$</td>
<td>246.4</td>
</tr>
<tr>
<td>$\delta_{out}$</td>
<td>4.9</td>
</tr>
<tr>
<td>$L_{plane, total}$</td>
<td>311.4</td>
</tr>
<tr>
<td>$\theta_{hopper}$</td>
<td>30°</td>
</tr>
<tr>
<td>$L_{stb}$</td>
<td>103.8</td>
</tr>
<tr>
<td>$\theta_{plane}$</td>
<td>27.3°</td>
</tr>
</tbody>
</table>
Carbobead CP 30/60 particles were poured into the hopper and released onto the plane. The particles traveled an initial length $L_{stb}$ prior to the analysis to ensure a fully developed, stable, unidirectional flow. A high-speed camera (Photron SA3) mounted perpendicular to the flow surface captured images at the highlighted region characterized by length $L_{PIV}$ using 1000 frames/s. PIV software (PIVLab) was used to compute free surface velocities using consecutive frames of 6 ms time intervals.

2.3.2 Model Contact Mechanics

The granular flow experiment on an inclined plane was modeled using LIGGGHTS. The different collision and rolling friction models used in LIGGGHTS for the DEM model in this study are listed in Table 2.2 [40]. Particle force interactions are determined via the Hertz spring-dashpot model, represented as:

$$F = (k\delta_{\text{overlap}} - \gamma_{\text{damp}} V_{\text{particle}})_n + (k\delta_{\text{overlap}} - \gamma_{\text{damp}} V_{\text{particle}})_t$$

(2.11)

where $k$ is an elastic constant; $\delta_{\text{overlap}}$ is the particle overlap between two interacting particles; $\gamma_{\text{damp}}$ is a viscoelastic damping constant; and $V_{\text{particle}}$ is the particle velocity. The elastic and viscoelastic damping constants are related to the mechanical properties of the particle. The tangential force is truncated to fulfill a static relation, given as:

$$F_t \leq \mu_s F_n$$

(2.12)

where $F_n$ is the normal force and $\mu_s$ is the coefficient of static friction.
Four rolling friction models are available in LIGGGHTS. However, the Elastic-Plastic-Spring-Dashpot (EPSD) and EPSD3 models were not considered as both require the coefficient of rolling viscous damping [23, 52], an iteratively-determined parameter obtained by trial and error [53]. The elastic-plastic-spring-dashpot 2 (EPSD2) and the constant directional torque (CDT) rolling friction models were considered as both require the empirically-determined coefficient of static rolling friction [40] for comparison. The rolling friction torque in the angular direction using the Constant-Direction-Torque (CDT) model is given as:

\[
\tau_{CDT} = \mu_r k_n \delta_n R_e \frac{\vec{\omega}_{r,s}}{||\vec{\omega}_{r,s}||},
\]

(2.13)

where \(\omega_{r,s}\) is the projection of the relative angular velocity between two interacting particles into the shear plane; and \(R_e\) is the equivalent radius of the two interacting particles, given as:

\[
\frac{1}{R_e} = \frac{1}{R_i} + \frac{1}{R_j},
\]

(2.14)

where \(i\) and \(j\) are the \(i^{th}\) and \(j^{th}\) particle, respectively. The rolling friction torque in the EPSD2 model is determined incrementally over multiple time steps using an elastic spring moment with no viscous damping contribution, given as:

\[
\tau_{EPSD2,t+\Delta t} = \tau_{EPSD2,t} - (k_t R_e) \Delta \theta_p,
\]

(2.15)
where $\Delta \theta_p$ is the incremental relative rotation between two interacting particles; $\tau_{\text{EPSD2},t}$ is the torque from the previous time step; and $\tau_{\text{EPSD2},t+\Delta t}$ has a set maximum controlled by $\mu_r$, given by:

$$\left| \tau_{\text{EPSD2},t+\Delta t} \right| \leq \mu_r R_e F_n,$$

(2.16)

<table>
<thead>
<tr>
<th>Model type</th>
<th>Contact Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Hertz</td>
<td>Elastic-plastic, spring, dashpot model used to calculate normal and tangential forces</td>
</tr>
<tr>
<td>Tangential</td>
<td>History</td>
<td>Indicates the inclusion of the tangential overlap spring force</td>
</tr>
<tr>
<td>Rolling Friction</td>
<td>EPSD2</td>
<td>Alternative elastic-plastic, spring, dashpot model used to impart a moment on every particle based upon the tangential overlap between particles.</td>
</tr>
<tr>
<td>Rolling Friction</td>
<td>CDT</td>
<td>A model that applies a constant torque to a particle related to the normal particle elasticity and overlap, and the relative rotational velocity between the particles</td>
</tr>
</tbody>
</table>

Table 2.2 A summary of LIGGGHTS contact models used to define the physical interaction between particles and their surroundings.

2.3.3 Model Development

The inclined flow model developed with LIGGGHTS represents an optimized version of various simulations that were completed to determine the most accurate and computationally efficient flow model of dense, inclined granular flow. Model development began with an investigation of the flow properties that exhibited the most influence on the flow behavior. A simplified inclined plane sensitivity study was conducted for each of the mechanical properties, aside from the coefficient of rolling friction. The velocity profile of the simplified flow is shown in Figure 2.6 including (a) a front view, and (b) a side view.
of the flow. A granular flow of spherical particles along the inclined plane was modeled with dimensions $80 \times 410 \, \text{mm}^2$ and angled at $31^\circ$ from the horizontal. An initial mass flow rate of 1.20 kg/s was specified for the inlet conditions, and particles were inserted in a continuous stream above the incline.

![Figure 2.6](image)

Figure 2.6 The flow geometry used in the sensitivity study of the mechanical properties. The average velocity magnitude within the region of interest was used as the primary output variable.

Once the flow reached steady state, the averaged particle velocity magnitudes were calculated for each case and the results can be seen in Table 2.3, where $d_p$ represents the particle diameter. Overall, the study indicated that the average flow velocity at steady state varies little when the mechanical properties are changed by $\pm 20\%$ of their mean value. The coefficient of friction and particle size were the most influential variables on the average velocity magnitude within the flow, due to the competing influence between free surface shear stress and particle / slope frictional interaction [54, 55]. The least influential variables
were the elastic properties because modeled particles include a high hardness, which resulted in repeatable collisional behavior.

Table 2.3 Modeling parameters and resulting average velocity of a sensitivity study on dense granular flows along an inclined plane

<table>
<thead>
<tr>
<th>Value</th>
<th>E, GPa</th>
<th>ν, -</th>
<th>μs, -</th>
<th>ε, -</th>
<th>dp, μm</th>
<th>V, m/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>90</td>
<td>0.15</td>
<td>0.6</td>
<td>0.45</td>
<td>500</td>
<td>0.801</td>
</tr>
<tr>
<td>E</td>
<td>101</td>
<td>0.15</td>
<td>0.6</td>
<td>0.45</td>
<td>500</td>
<td>0.792 (-1.13%)</td>
</tr>
<tr>
<td>ν</td>
<td>72</td>
<td>0.15</td>
<td>0.6</td>
<td>0.45</td>
<td>500</td>
<td>0.797 (-0.521%)</td>
</tr>
<tr>
<td>μs</td>
<td>90</td>
<td>0.18</td>
<td>0.6</td>
<td>0.45</td>
<td>500</td>
<td>0.808 (+0.774%)</td>
</tr>
<tr>
<td>ε</td>
<td>90</td>
<td>0.12</td>
<td>0.6</td>
<td>0.45</td>
<td>500</td>
<td>0.796 (-0.678%)</td>
</tr>
<tr>
<td>dp</td>
<td>90</td>
<td>0.15</td>
<td>0.72</td>
<td>0.45</td>
<td>500</td>
<td>0.785 (-5.39%)</td>
</tr>
</tbody>
</table>

The study reinforced the need to develop models that accurately capture friction and particle size effects. Therefore, a non-spherical particle model of the inclined flow was developed to represent the shape of Carbobead CP 30/60 particles. Particles were modeled as ellipsoids and were inserted as a packed bed into the flow geometry seen in Figure 2.5. These modifications enabled the simulation of dense granular flows, however, the ellipsoidal particles caused both a severe over-prediction of the effects of friction, due to larger surface area, and a significantly increased computation time, due to the tracking of particle orientation.

Spherical particle models that account for particle rolling friction have been shown to accurately account for both particle shape and friction effects [36, 56], therefore, a spherical particle model coupled with extrinsic μt was used to decrease the computation time and to account for particle shape in the final model. An 8 mm wide flow region at the
The center of the inclined plane and hopper shown in Figure 2.5b, was used as the modeling domain. The boundaries across the width of this region were modeled as frictionless walls to capture particle-particle collisional effects at the boundary while eliminating shear effects. A $100 \times 8 \text{ mm}^2$ overlap region between PIV measurements and DEM model, characterized by $L_{\text{PIV}}$ and $w_{\text{DEM}}$ respectively, was used for comparison.

The DEM modeling was performed by considering a total of 187,890 particles that were inserted within the hopper with 50% volume fraction with no initial velocity. The particle sizes were modeled using the measured $d_{\text{eff}}$ distributions.

2.4 Results and Discussion

The flow properties are provided, and the results from the experiment and DEM model are discussed and compared for the temporal granular flow propagation on an inclined plane. Validation of the model results for the transient and underdeveloped regime as well as steady and developed regime are given.

2.4.1 Particle Shape and Size Distributions

Optical microscopy of Carbobead CP particles was performed for 60/100, 40/100, 30/60, and 50/140 US standard mesh sizes. A Carbobead CP 30/60 particle image is shown in Figure 2.6. Particle shape and size distributions were calculated using an image processing algorithm that identified particle boundaries from the contrasting background.
Figure 2.7 Optical microscopy image of Carbobead CP 30/60 particles from an optical microscope (Leica, USA).

Results for particle size characterization of Carbobead CP 30/60 from histograms are shown in Figure 2.7 for (a) $d_{\text{eff}}$, (b) $d_{\text{major}}$, and (c) $d_{\text{minor}}$. Different probability distribution functions (pdf) were used to describe the measurements. The $d_{\text{eff}}, d_{\text{major}}, d_{\text{minor}}$, were approximated with a lognormal pdf, given as:

$$f(x) = \frac{1}{x\omega\sqrt{2\pi}} \exp\left[\frac{-(\ln(x) - \phi)^2}{2\omega^2}\right], \quad (2.17)$$

where the parameters $\phi$ and $\omega^2$ are the normal distribution mean and variance, respectively. The expected value and variance for the lognormal distribution were computed, respectively, as:

$$E(x) = \exp\left(\phi + \frac{\omega^2}{2}\right), \quad (2.18)$$

$$V(x) = \exp(2\phi + \omega^2) [\exp(\omega^2) - 1], \quad (2.19)$$
Results for particle shape characterization of Carbobead CP 30/60 from histograms are shown respectively in Figure 2.8 for (a) C, (b) R, and (c) AR. A lognormal distribution was fit to AR, while C and R pdfs were approximated with Weibull distribution, given as:

\[ f(x) = \frac{\gamma}{\kappa} \left( \frac{x}{\kappa} \right)^{\gamma-1} \exp \left[ - \left( \frac{x}{\kappa} \right)^{\gamma} \right], \tag{2.20} \]

where \( \gamma \) is the shape parameter and \( \kappa \) is the scale parameter. The expected values and variance were computed, respectively, as:

\[ E(x) = \kappa \Gamma \left( 1 + \frac{1}{\gamma} \right), \tag{2.21} \]
\[ V(x) = \kappa^2 \Gamma \left( 1 + \frac{2}{\gamma} \right) - \kappa^2 \left[ \Gamma \left( 1 + \frac{1}{\gamma} \right) \right]^2, \tag{2.22} \]

The statistical parameters for particle shape and size distributions for Carbobead CP 30/60 are listed in Table 2.3. The distributions for \( d_{\text{eff}}, d_{\text{major}}, d_{\text{minor}}, \) and AR were lognormal, whereas, C and R followed the Weibull distribution.

Shape and size parameters for Carbobead CP particles with mesh sizes 50/140, 40/100, and 30/60 were measured and compared, and the means and standard deviations are reported in Table 2.4. No significant changes were observed due to size; however, the Carbobead CP 50/140 particles have a slightly higher AR compared to the larger particles. The measured median \( d_{\text{eff}} \) and \( d_{\text{minor}} \) were within 5% of \( d_{\text{eff}} \) and \( d_{\text{minor}} \), respectively, for all mesh sizes and, therefore, assumed to be normally distributed.
Particle shape and size parameters were compared with technical datasheets for Carbobead particles from Carbo Ceramics Inc. The reported medians of the particle size for 50/140 and 30/60 were 191 and 453 µm, respectively, which were higher than $d_{\text{eff}}$, but closer to $d_{\text{minor}}$ for the respective mesh sizes. Roundness and sphericity were reported as 0.9 each from a visual estimation using the Krumbein and Sloss method [57]. The shape parameters were higher than the measured $C$ and $R$.

Table 2.4 Probability distribution function parameters representative of measured particle shape and size data for Carbobead CP 30/60.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Distribution</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_{\text{eff}}$</td>
<td>lognormal</td>
<td>$\varphi = 6.04, \omega = 0.14$</td>
</tr>
<tr>
<td>$d_{\text{major}}$</td>
<td>lognormal</td>
<td>$\varphi = 6.30, \omega = 0.16$</td>
</tr>
<tr>
<td>$d_{\text{minor}}$</td>
<td>lognormal</td>
<td>$\varphi = 6.14, \omega = 0.14$</td>
</tr>
<tr>
<td>$C$</td>
<td>Weibull</td>
<td>$\gamma = 20.60, \kappa = 0.75$</td>
</tr>
<tr>
<td>$R$</td>
<td>Weibull</td>
<td>$\gamma = 17.24, \kappa = 0.86$</td>
</tr>
<tr>
<td>$AR$</td>
<td>lognormal</td>
<td>$\varphi = 6.04, \omega = 0.14$</td>
</tr>
</tbody>
</table>
Figure 2.8 Particle size characterization of Carbobead CP 30/60 particles from histograms with fitted lognormal probability distribution functions for (a) effective diameter, (b) major diameter, and (c) minor diameter.
Figure 2.9 Particle shape characterization of Carbobead CP 30/60 particles from histograms with fitted probability distribution functions for (a) circularity (b) roundness, and (c) aspect ratio
Table 2.5 Statistical shape and size parameters for Carbobead CP particles for various US standard mesh sizes.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>50/140</th>
<th>40/100</th>
<th>30/60</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_{\text{eff}}, \mu$m</td>
<td>167 ± 31.6</td>
<td>230 ± 38.3</td>
<td>418 ± 58.9</td>
</tr>
<tr>
<td>$d_{\text{major}}, \mu$m</td>
<td>211 ± 47.4</td>
<td>298 ± 53.5</td>
<td>549 ± 88.9</td>
</tr>
<tr>
<td>$d_{\text{minor}}, \mu$m</td>
<td>173 ± 35.7</td>
<td>258 ± 44.5</td>
<td>470 ± 65.8</td>
</tr>
<tr>
<td>C, –</td>
<td>0.81 ± 0.05</td>
<td>0.73 ± 0.07</td>
<td>0.73 ± 0.06</td>
</tr>
<tr>
<td>R, –</td>
<td>0.78 ± 0.08</td>
<td>0.83 ± 0.07</td>
<td>0.83 ± 0.07</td>
</tr>
<tr>
<td>$AR$, –</td>
<td>1.22 ± 0.13</td>
<td>1.16 ± 0.13</td>
<td>1.17 ± 0.09</td>
</tr>
</tbody>
</table>

2.4.2 Elastic Properties

The fitted elastic properties as a function of $p$ fit to Spriggs’ correlations are shown in Figure 2.9 for (a) $E$, (b) $G$, and (c) $v$. $p = 0$ corresponds to a $\rho_t = 3.54$ g/cm$^3$ (data from Carbo Ceramics Inc.) for Carbobead CP Flour. All the elastic properties increased with increases in $\rho_s$ and decreases in $p$ for different specimens. The vertical dashed line represents $\rho = 3.27$ g/cm$^3$ of Carbobead CP [13]. The $\bar{\rho}$ of intermediately dense specimens matched closely with $\rho_t$ of Carbobead CP particles. Fitted parameters for the Spriggs’ correlations are listed in Table 2.5. The means and standard deviations of elastic properties for Carbobead CP specimen hot-pressed to densities equivalent to Carbobead CP particles are listed in Table 2.6. The measurements were precise with small standard deviations.

Table 2.6 Mean and standard deviation of elastic properties for Carbobead CP specimen hot-pressed to densities equivalent to Carbobead CP particles

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_s$, g/cm$^3$</td>
<td>3.28 ± 0.0369</td>
</tr>
<tr>
<td>$E$, GPa</td>
<td>209 ± 5.51</td>
</tr>
<tr>
<td>$G$, GPa</td>
<td>81.6 ± 3.05</td>
</tr>
<tr>
<td>$v$, –</td>
<td>0.28 ± 0.033</td>
</tr>
</tbody>
</table>
Figure 2.10 Elastic properties of Carbobead CP with Spriggs’ fit to (a) elastic modulus, (b) shear modulus, and (c) Poisson’s ratio as a function of material porosity.
Table 2.7 Fitted parameters for elastic properties Spriggs’ correlations of Carbobead CP

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_0$, GPa</td>
<td>288.84</td>
</tr>
<tr>
<td>$G_0$, GPa</td>
<td>111.19</td>
</tr>
<tr>
<td>$\nu_o$, -</td>
<td>3.07</td>
</tr>
<tr>
<td>$b_e$, -</td>
<td>5.11</td>
</tr>
<tr>
<td>$b_g$, -</td>
<td>4.63</td>
</tr>
</tbody>
</table>

2.4.3 Coefficient of Restitution

Impact tests for Carbobead CP particles were conducted with three standard size ranges: (1) 50/140, (2) 40/100, and (3) 30/60. The results were averaged together to approximate the behavior of single particle impacts for each given size range shown in Table 2.7. The calculated $\bar{\varepsilon}$ showed no significant change between particle sizes, indicative of an intrinsic property. One-way analysis of variance (ANOVA) testing was conducted between the three measurements sets for different sizes, resulting in insufficient evidence of different means ($p = 0.736$).

Table 2.8 Mean coefficient of restitution for 10 Carbobead CP particle-to-particle collisions and standard deviation for separate US standard mesh size ranges

<table>
<thead>
<tr>
<th>Mesh size range</th>
<th>$\varepsilon$, -</th>
</tr>
</thead>
<tbody>
<tr>
<td>50/140</td>
<td>0.496 ± 0.0840</td>
</tr>
<tr>
<td>40/100</td>
<td>0.524 ± 0.100</td>
</tr>
<tr>
<td>30/60</td>
<td>0.519 ± 0.0659</td>
</tr>
</tbody>
</table>

2.4.4 Model Validation with Experimental Flow Results

The flow properties used in DEM model inputs are listed in Table 2.8. Properties reported in literature for $E$ and $\nu$ for alumina wall [58] and $\nu$ for the resin-based hopper [59] were used. The $E$ for hopper was obtained from the manufacturer. The wall-particle properties, including $\varepsilon$, $\mu_s$, and $\mu_r$, were measured for both the painted alumina board and
hopper using the developed methods described in Section 2. A surface glued with Carbobead CP 30/60 particles was used as the top sliding surface for the wall-particle $\mu_s$ measurements to account for the effect of large surface roughness of the painted alumina surface. The coefficients of static sliding, $\mu_s$, and static rolling, $\mu_r$, were measured by a colleague.

Table 2.9 Mechanical properties used in the simulation of inclined flow for Carbobead CP 30/60 particulates

<table>
<thead>
<tr>
<th>Material property</th>
<th>Particle</th>
<th>Alumina wall</th>
<th>Hopper</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$, GPa</td>
<td>209</td>
<td>360</td>
<td>2.80</td>
</tr>
<tr>
<td>$\nu$, –</td>
<td>0.28</td>
<td>0.25</td>
<td>0.38</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contact Property</th>
<th>Particle-particle</th>
<th>Alumina-particle</th>
<th>Hopper-particle</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$, –</td>
<td>0.52</td>
<td>0.41</td>
<td>0.70</td>
</tr>
<tr>
<td>$\mu_s$, –</td>
<td>0.53</td>
<td>1.00</td>
<td>0.80</td>
</tr>
<tr>
<td>$\mu_r$, –</td>
<td>0.37</td>
<td>0.41</td>
<td>0.40</td>
</tr>
</tbody>
</table>

The mean, local free surface velocities were computed from particle displacements between successive images measured with PIV for both transient and steady flow regimes. The images were divided into small sub-images or interrogation spots of $24 \times 24$ pixel$^2$, each providing the mean local displacement of the particles. The mean local velocities were computed from the magnification of the images (in pixel/mm) and the delay between the two images (in s).

Results from the PIV analysis from a $100 \times 8$ mm$^2$ region of interest for different times are shown in Figure 2.10 for (a) 2.38 s, (b) 2.75 s, and (c) 5 s. The flows at $t = 2.38$ s and $t = 2.75$ s were transient and underdeveloped, and a steady and fully developed granular flow was observed at $t = 5$ s. The transient flows were characterized by dense middle regions moving at lower velocities and dispersed, high velocity regions occurring
further downstream. The vectors seen in orange in Figure 2.10a represent interpolated velocities computed as result of particle scarcity at those locations. The flows were more unidirectional and stable nearing steady conditions.

The free-surface velocity fields in the DEM model using the constant directional torque (CDT) and the alternative elastic-plastic-spring-dashpot (EPSD2) rolling friction models were compared with measured velocity fields in Figure 2.11 for (a) \( t = 2.38 \) s, (b) \( t = 2.75 \) s, and (c) \( t = 5 \) s. \( L_p \) represents the distance along the flow direction within the PIV measurement region as shown in Figure 2.5b. Each velocity vector from the model and experiment represents the local average free-surface velocity measured from respective discrete regions along the length of the inclined plane. The PIV measurements uncertainties, which included systematic, magnification, and length scale measurement error, were determined prior to measurements from calibration of the PIV measurement system for prescribed particle velocities using a conveyor belt. Ninety-five percent confidence intervals were determined from experimental error determined from repeated calibration measurements of the PIV measurement system compared to prescribed particle velocities according to a normal distribution. The increase in measured velocity seen for \( t = 2.38 \) and 2.75 s correspond to the dispersed, high-velocity flows. At \( t = 2.38 \) s and \( L_p > 80 \) mm, an expected increase in velocity was not observed due to imprecisions from interpolated PIV.

The CDT model in all three cases captured the overall flows of particles significantly better than the EPSD2 model. The EPSD2 model over-predicted velocity in the transient regime while systematically under-predicting the velocities during steady flow. The dispersed, high-velocity flow regions for the EPSD2 cases were further upstream
compared to the experiment, clearly indicating a time lag in flow propagation. At $t = 5$ s, an overall decrease of flow velocity was observed. These observations indicated the coefficient of static rolling friction was inadequately considered in the EPSD2 model. The CDT model demonstrated receding dispersed regions characterized by decreased downstream velocities from $t = 2.38$ s to $t = 5$ s, capturing the physics of flows approaching steady state. An increase in accuracy of the CDT model was clearly observed as flow approached steady conditions. The velocity vectors from the CDT model at $t = 5$ s were within 95% confidence interval of the measured velocities. The average difference between the experiment and CDT model was $<10%$. 
Figure 2.11 Images of PIV analysis on granular flow on an inclined plane with surface velocity vectors on a 100 × 8 mm² region of interest for different times after particle release including (a) 2.38 s, (b) 2.75 s, and (c) 5 s.
Figure 2.12 Free-surface velocities of particles along a 100 × 8 mm² region of interest on an inclined slope as a function of distance from the hopper along flow direction obtained using CDT and EPSD2 models from LIGGGHTS and measured velocities using PIV for different times after particle release including (a) 2.38 s, (b) 2.75 s, and (c) 5 s.
2.5 Conclusions

Granular flows of sintered bauxite proppants were examined along an inclined plane for application of thermal energy storage for particle-based concentrated solar power receivers and reactors. Flow properties necessary to accurately model granular flows were measured for Carbobead CP 50/140, 40/100, and 30/60 particles. The shape distribution of particle was nearly spherical, and particle size did not have a significant impact on the coefficients of restitution. The particle shape distribution resulted in high variability in static rolling friction measurements.

An experiment with a granular flow of Carbobead CP 30/60 particles along an inclined plane was compared to results from numerical models in LIGGGHTS employing the discrete element method. The free surface velocities measured by particle image velocimetry were compared with modeled velocities. The alternative elastic-plastic-spring-dashpot rolling friction model did not adequately consider the effects of static rolling friction and resulted in a time lag in flow propagation compared to the experiment. The constant directional torque rolling friction model produced good results for flows in both transient and steady regimes. A further improvement in accuracy was observed as the flow approached steady state, where the free surface velocities from the numerical model were within the 95% confidence intervals of the measured free surface velocities in the experiment (average velocity difference of <10%).

Note: The experimental data for the room temperature flow properties can be downloaded from: http://hdl.handle.net/1853/62903
CHAPTER 3. HIGH TEMPERATURE FLOW CHARACTERIZATION

3.1 Introduction

The measurement of Carbobead CP sintered bauxite flow properties at elevated temperatures was completed, and an investigation of the associated impact on granular flows with DEM models are presented. Granular flow modeling along inclined plane was performed to determine the influence of temperature on the flow behavior. High temperature flow properties of Carbobead CP particles were measured, including particle shape and size distributions; the elastic and shear moduli and Poisson’s ratio; the coefficients of static sliding and static rolling friction; and the coefficient of restitution up to 800 °C. The DEM models using the flow properties at different temperatures were compared at steady state, and the resulting velocity fields, volume fraction, particle bed thickness, average velocity magnitude, and mass flux contours were calculated [60].

3.2 Measured Flow Properties in the DEM

The DEM is a Lagrangian numerical technique used to model granular flows [61], utilizing particle mechanical properties to predict the flow behavior. As particles approach one another, contact mechanics are used to determine particle trajectory by calculating local frictional and collisional forces. Measured mechanical properties of the particulate are used as inputs and are required to develop accurate, robust simulations that are predictive of experimental granular flow behavior [36, 42, 43]. Measured mechanical properties (e.g., particle shape and size, elastic properties, static friction coefficients and
coefficient of restitution) have enabled precise simulation of particle-particle and particle-surface interactions with a high degree of accuracy [40, 41, 43]. The DEM models presented in this work were developed using LIGGGHTS [LAAMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) Improved for General Granular and Granular Heat Transfer Simulations] v-3.8.0, an open-source DEM simulation software [40]. DEM models of Carbobead CP particle flows along an inclined plane at several temperatures were developed to better inform the design and modeling of next generation SPHRR.

3.2.1 Particle Contact Mechanics

Particle behavior was determined using contact laws that define the physics of particle interactions with particle mechanical properties. Contact models in LIGGGHTS are used to capture normal, tangential, and angular contacts. For the normal and tangential directions, the Hertz model and the history tangential contact models were used [23, 40, 43]. The forces incident on interacting particles in the models are determined via a spring-dashpot model, as shown in Figure 3.1, and represented in Equations 2.11 and 2.12.

The elastic and viscoelastic damping constants are related to the mechanical properties of the particle. They are determined through effective material properties, which represent a combination of the mechanical properties of each interacting particle, represented, respectively, as:

\[
\beta = \frac{\ln(\varepsilon)}{\sqrt{\ln^2(\varepsilon) + \pi^2}},
\]

(3.1)
\[
\frac{1}{E_e} = \frac{1 - \nu_i^2}{E_i} + \frac{1 - \nu_j^2}{E_j},
\]

(3.2)

\[
\frac{1}{G_e} = \frac{2(2 - \nu_i)(1 + \nu_i)}{E_i} + \frac{2(2 - \nu_j)(1 + \nu_j)}{E_j},
\]

(3.3)

where \(E_e\) is the effective elastic modulus; \(G_e\) is the effective shear modulus; \(\varepsilon\) is the coefficient of restitution; \(\nu\) is the Poisson’s ratio; \(\beta\) is a material constant used in calculating \(k\) and \(\gamma\); and subscripts \(i\) and \(j\) denote the two interacting particles, respectively.

Figure 3.1 A spring-dashpot model describing particle interactions in the normal and tangential directions.

For the angular direction, rolling friction torque was used with the coefficient of static rolling friction. Four rolling friction models are available for implementation in LIGGGHTS, however, only the Constant-Direction-Torque (CDT) model was used based on previous work that showed superior performance for inclined granular flows [43]. The
CDT torque is determined with the relative angular velocity between interacting particles, represented in Equation 2.16.

The contact models implemented required the input of particle mechanical properties. DEM models that utilize physically measured mechanical properties have enabled the realization of physically-significant flow behavior across different geometries [40, 43]. Several works have substituted some of these mechanical properties with calibrated properties to reduce computation time, or in the absence of measured properties [23, 62-64]. However, the scope of calibrated DEM models has remained limited as the extension of these models to other geometries is not possible without additional model calibration [36, 42, 43]. The implementation of measured particle properties in DEM models has provided a more robust pathway for determining particle flow behavior.

3.2.2 Experimental Measurements of Particle Properties

A series of measurements of sintered bauxite particles was completed for temperatures up to 800 °C for the development of DEM models based on measured flow properties. The dominant particle properties that influence the flow were measured, including particle shape and size distributions, elastic properties, coefficients of static sliding and static rolling friction, and coefficient of restitution.

3.2.2.1 Temperature-dependence of particle shape and size

The effect of temperature on the particle shape and size distributions were predicted using the linear thermal expansion coefficient obtained from the manufacturer as $\alpha_1 =$
2.34 \cdot 10^{-6} \text{ mm} \cdot \text{ mm}^{-1} \cdot \text{ K}^{-1}. The predicted change in perimeter and projected area as a function of temperature and ambient measurements [43] are given, respectively, as:

\[ \Delta P = P_0 \alpha_l \Delta T, \]  \hfill (3.4)
\[ \Delta A = 2A_0 \alpha_l \Delta T, \]  \hfill (3.5)

where \( P_0 \) and \( A_0 \) are the particle perimeter and projected area at ambient temperature, respectively [65]; and \( \Delta T \) is the temperature change. The changes due to thermal expansion in major particle diameter, \( d_{\text{major}} \), and minor particle diameter, \( d_{\text{minor}} \), were calculated directly using the linear thermal expansion relationships. Negligible differences due to thermal expansion were calculated for each distribution, and only a 0.18% increase in the average effective diameter was observed at 800 °C.

3.2.2.2 Elastic Properties

The elastic properties of Carbobead CP were measured using the impulse excitation method with hot-pressed specimens of the particles [48]. The measurements were adjusted to Carbobead CP particle density using the Sprigg’s correlations determined at ambient temperatures [43, 49, 50].

High temperature elastic properties with density corrections of 10 hot-pressed Carbobead CP specimens are shown in Figure 3.2 for (a) the elastic and shear moduli and (b) Poisson’s ratio. Both the elastic and shear moduli decreased with increasing temperature while the Poisson’s ratio increased gradually at higher temperatures, due to the elastic modulus decreasing slower than twice the decrease in shear modulus. The trends
agreed with measurements in similar aluminosilicate materials [58, 66-68]. The average elastic properties at temperatures of interest are listed in Table 3.1.

Figure 3.2 Measured elastic properties of Carbobead CP as a function of temperature with (a) the elastic and shear modulus, and (b) Poisson’s ratio.
Table 3.1 The average, scaled high temperature elastic properties of Carbobead CP particles at temperatures of interest.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$E$, GPa</th>
<th>$G_s$, GPa</th>
<th>$v$, -</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>209</td>
<td>81.6</td>
<td>0.283</td>
</tr>
<tr>
<td>200</td>
<td>206</td>
<td>80.4</td>
<td>0.283</td>
</tr>
<tr>
<td>400</td>
<td>202</td>
<td>78.8</td>
<td>0.285</td>
</tr>
<tr>
<td>600</td>
<td>197</td>
<td>76.8</td>
<td>0.286</td>
</tr>
<tr>
<td>800</td>
<td>191</td>
<td>74.0</td>
<td>0.289</td>
</tr>
</tbody>
</table>

3.2.2.3 Coefficient of Restitution

The coefficient of restitution ($\varepsilon$) of Carbobead CP 30/60 was measured as a function of temperature. Other size distributions were not measured, as $\varepsilon$ was shown not to significantly change with particle size [43]. Schematics of the experimental setup are shown in Figure 3.3 for (a) the insulated alumina enclosure and (b) of the particle drop and interchangeable impact plane mechanisms. The particles and the impact plane were heated via the heater and tubular cartridge heaters prior to dropping. Once the desired temperature was achieved, the particle knife valve was opened to allow particles to flow, and particle motion was captured with a high-speed camera (Photon SA3, $512 \times 768$ pixel$^2$) at a rate of 2000 frames/s.

Both the particles and the impact plane were heated to ensure a controlled, repeatable impact temperature. The temperatures of the impact plane, the surrounding air, and the particles before release were all recorded with K-type thermocouples. To provide an estimate of particle temperature at impact, the heat loss from the particles while falling
was estimated using a lumped capacitance heat transfer model of the particle with simplified radiative heat transfer. The convective heat transfer to the surroundings was estimated using the Ranz and Marshall correlation, given as:

$$\overline{Nu}_D = 2 + 0.6 \text{Re}_D^{1/2} \text{Pr}^{1/3}, \quad (3.6)$$

where $\text{Pr}$ is the Prandtl number and $\text{Re}_D$ is the Reynolds number of the air surrounding the particle, given as:

$$\text{Re}_D = \frac{V_{\text{before}} d_p}{\nu_f}, \quad (3.7)$$

where $V_{\text{before}}$ is the magnitude of the velocity before impact, $d_p$ is the particle diameter and $\nu_f$ is the kinematic viscosity of the air [69].

The radiative heat transfer to the particle was determined assuming a constant temperature heat source. The view factor from the heater to the particle was determined using coaxial parallel disks despite some slight geometric differences, given as:

$$F_{ij} = \frac{1}{2} \left( S - \left[ S^2 - 4 \left( \frac{r_j}{r_i} \right)^2 \right]^{1/2} \right), \quad (3.8)$$

$$S = 1 + \frac{1 + R_j^2}{R_i^2}; \quad R_i = \frac{r_h}{l}; \quad R_j = \frac{r}{l}, \quad (3.9)$$

where $r_h$ is the effective radius of the heater; $r$ is the particle radius; and $l$ is the perpendicular distance between the heater and the particle.
Figure 3.3 Coefficient of restitution high temperature experimental setup with a schematic of (a) the alumina enclosure and (b) the particle drop and impact plane mechanisms.

The $\epsilon$ for Carbobead CP 30/60 particle-particle impacts as a function of temperature are shown in Figure 3.4. No significant changes in the $\epsilon$ were observed as temperature increased for particle-particle and particle-alumina impacts, however, an overall decrease
in the $\varepsilon$ was observed with increasing temperature for all impacts, consistent with previous work [70, 71]. Convective heat losses appeared to play a significant influence on particle temperature at impact. The mean and standard deviations of $\varepsilon$ and corresponding system temperatures for particle-particle and particle-alumina impacts are listed in Table 3.2 and Table 3.3.

![Figure 3.4](image)

Figure 3.4 Measured coefficient of restitution for Carbobead CP 30/60 particle-particle impacts as a function of temperature, where ‘+’ indicates outliers.

Table 3.2 Mean and standard deviation of the coefficient of restitution of Carbobead CP 30/60 particle-particle impacts with corresponding system temperatures.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$\varepsilon$, -</th>
<th>$T_{p,\text{drop}}$, °C</th>
<th>$T_{\text{impact}}$, °C</th>
<th>$T_{p,\text{impact}}$, °C</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>0.531 ± 0.0612</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>200</td>
<td>0.512 ± 0.0958</td>
<td>239</td>
<td>215</td>
<td>156</td>
</tr>
<tr>
<td>400</td>
<td>0.501 ± 0.0816</td>
<td>473</td>
<td>409</td>
<td>305</td>
</tr>
<tr>
<td>600</td>
<td>0.461 ± 0.0861</td>
<td>646</td>
<td>623</td>
<td>532</td>
</tr>
<tr>
<td>800</td>
<td>0.447 ± 0.0804</td>
<td>868</td>
<td>756</td>
<td>608</td>
</tr>
</tbody>
</table>
Table 3.3 Mean and standard deviation of the coefficient of restitution of Carbobead CP 30/60 particle-alumina impacts with corresponding system temperatures.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$\varepsilon$, -</th>
<th>$T_{p,\text{drop}}$, °C</th>
<th>$T_{\text{impact}}$, °C</th>
<th>$T_{p,\text{impact}}$, °C</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>0.522 ± 0.0822</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>200</td>
<td>0.530 ± 0.0894</td>
<td>256</td>
<td>210</td>
<td>163</td>
</tr>
<tr>
<td>400</td>
<td>0.502 ± 0.1205</td>
<td>483</td>
<td>402</td>
<td>246</td>
</tr>
<tr>
<td>600</td>
<td>0.491 ± 0.0968</td>
<td>630</td>
<td>592</td>
<td>349</td>
</tr>
<tr>
<td>800</td>
<td>0.487 ± 0.0956</td>
<td>803</td>
<td>806</td>
<td>608</td>
</tr>
</tbody>
</table>

3.3 Particle Flow Model Development

Granular flows along an inclined plane were modeled with LIGGGHTS to determine the influence of temperature on the flow de-coupled from heat transfer. DEM models were developed for flows at 23 °C, 200 °C, 400 °C, 600 °C, and 800 °C.

3.3.1 Model Mechanical Properties and Neighbour Pairing

The flow input properties used for each of the DEM models are listed in Table 3.4 and Table 3.5 and were used to calculated the forces between particles. Rather than calculating forces between every particle pair, the frictional and collisional forces between a pair of particles was only calculated when the particles had the potential to interact. Lists of particle pairs with the potential to interact were output periodically throughout the simulation. An interacting pair of particles was only included in the list if the particles were within a certain range of each other, given as:

$$\|\vec{x}_i - \vec{x}_j\| < r_i + r_j + s,$$  \hspace{1cm} (3.10)
where $\mathbf{x}$ is the position vector of a particle of interest; $r$ is the radius of a particle of interest; and $s$ is the freely-chosen, user-defined skin parameter [40]. As particles shifted and bounced to different locations in a system, interacting particles potentially separate with new particle pairs formed. Therefore, accurately capturing inter-particle forces required periodic updates to the particle pairs lists. The particle lists in the current simulations were rebuilt at every timestep after a particle in the list traveled a distance greater than half of $s = 235.6 \mu m$.

Table 3.4 Intrinsic and particle-particle mechanical properties of Carbobead CP 30/60 particles used in the simulation of Carbobead CP particle flows along an inclined plane.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$E$, GPa</th>
<th>$\nu$, -</th>
<th>$\varepsilon_s$, -</th>
<th>$\mu_s$, -</th>
<th>$\mu_t$, -</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>252</td>
<td>0.258</td>
<td>0.531</td>
<td>0.509</td>
<td>0.371</td>
</tr>
<tr>
<td>200</td>
<td>248</td>
<td>0.258</td>
<td>0.512</td>
<td>0.529</td>
<td>0.284</td>
</tr>
<tr>
<td>400</td>
<td>243</td>
<td>0.260</td>
<td>0.502</td>
<td>0.552</td>
<td>0.286</td>
</tr>
<tr>
<td>600</td>
<td>238</td>
<td>0.261</td>
<td>0.461</td>
<td>0.631</td>
<td>0.277</td>
</tr>
<tr>
<td>800</td>
<td>229</td>
<td>0.263</td>
<td>0.447</td>
<td>0.797</td>
<td>0.335</td>
</tr>
</tbody>
</table>

Table 3.5 Intrinsic and particle-wall mechanical properties of rigid alumina board used in the simulation of Carbobead CP particle flows along an inclined plane.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$E$, GPa</th>
<th>$\nu$, -</th>
<th>$\varepsilon_s$, -</th>
<th>$\mu_s$, -</th>
<th>$\mu_t$, -</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>360</td>
<td>0.250</td>
<td>0.522</td>
<td>1.016</td>
<td>0.403</td>
</tr>
<tr>
<td>200</td>
<td>352</td>
<td>0.253</td>
<td>0.530</td>
<td>1.110</td>
<td>0.323</td>
</tr>
<tr>
<td>400</td>
<td>343</td>
<td>0.256</td>
<td>0.502</td>
<td>1.045</td>
<td>0.297</td>
</tr>
<tr>
<td>600</td>
<td>335</td>
<td>0.260</td>
<td>0.401</td>
<td>0.958</td>
<td>0.322</td>
</tr>
<tr>
<td>800</td>
<td>327</td>
<td>0.263</td>
<td>0.487</td>
<td>0.948</td>
<td>0.356</td>
</tr>
</tbody>
</table>
3.3.2 Flow Geometry and Particle Insertion

A schematic of the inclined plane geometry is shown in Figure 3.5 with related dimensions listed in Table 3.6. The inclined plane was at an angle of 27.3° from the horizontal and made of rigid alumina board. A symmetrical boundary condition was used along the middle of the flow using a frictionless particle wall (denoted by Plane $A$) to reduce the computational time. Wall effects on the flow were also captured by defining a plane of the rigid alumina board along the right side of the flow (denoted by Plane $B$ with a hashed edge). Consistent particle flow inlet conditions were captured with two additional 2D planes, (denoted by the dark grey surfaces). These planes were modeled using the room-temperature mechanical properties of photopolymer resin. The flow entered the system at the particle inlet within the dark grey surfaces and propagated along the incline in the positive $x$-direction.

Particles were introduced to the system as a continuous stream. Particles were inserted in the $19.1 \times 25.4 \times 3.92$ mm$^3$ region (denoted by the variables $a \times w \times h$). Six distinct particle insertion regions with varying initial conditions were identified from the previously validated model and used to accurately capture the inlet flow behavior. The insertion regions are shown in Figure 3.6 with related input parameters listed in Table 3.7, where $C_{\text{min}}, C_{\text{max}}, \vec{V},$ and $\phi$ represent the minimum coordinates of the insertion region, the maximum coordinates of the insertion region, the particle velocity vectors defined at the inlet, and the particle volume fraction, respectively. Particles in Regions 1, 2 and 3 were periodically inserted to provide a continuous flow of particles, and particles in Regions 4, 5 and 6 were inserted only once at the beginning of the simulation. The polydisperse
particle size distributions of each layer were also included in the current study, listed in Table 3.8.

![Figure 3.5](image)

**Figure 3.5** The geometry in a numerical analysis of room and high temperature particle flows along an inclined plane.

**Table 3.6** List of dimensions used in the inclined flow geometry.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w$</td>
<td>25.4 mm</td>
</tr>
<tr>
<td>$L_1$</td>
<td>65.1 mm</td>
</tr>
<tr>
<td>$W$</td>
<td>40.0 mm</td>
</tr>
<tr>
<td>$L_2$</td>
<td>246.3 mm</td>
</tr>
<tr>
<td>$a$</td>
<td>19.1 mm</td>
</tr>
<tr>
<td>$h$</td>
<td>3.92 mm</td>
</tr>
</tbody>
</table>
Figure 3.6 The insertion regions used to define inlet conditions for optimized DEM models of room and high temperature particle flows along an inclined plane.

Table 3.7 Parameters used for particle insertion regions and particle inlet conditions.

<table>
<thead>
<tr>
<th>Region</th>
<th>$C_{\text{min}}$, m</th>
<th>$C_{\text{max}}$, m</th>
<th>$\bar{V}$, m/s</th>
<th>$\phi$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(0.0500, 0.0486, 0.0144)</td>
<td>(0.0520, 0.0504, 0.0398)</td>
<td>&lt; 0.0762, 0, 0 &gt;</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>(0.0480, 0.0479, 0.0144)</td>
<td>(0.0500, 0.0491, 0.0398)</td>
<td>&lt; 0.0492, 0, 0 &gt;</td>
<td>0.5</td>
</tr>
<tr>
<td>3</td>
<td>(0.0460, 0.0470, 0.0144)</td>
<td>(0.0480, 0.0479, 0.0398)</td>
<td>&lt; 0.0180, 0, 0 &gt;</td>
<td>0.188</td>
</tr>
<tr>
<td>4</td>
<td>(0.0520, 0.0491, 0.0144)</td>
<td>(0.0651, 0.0509, 0.0398)</td>
<td>&lt; 0.0762, 0, 0 &gt;</td>
<td>0.55</td>
</tr>
<tr>
<td>5</td>
<td>(0.0490, 0.0479, 0.0144)</td>
<td>(0.0651, 0.0491, 0.0398)</td>
<td>&lt; 0.0492, 0, 0 &gt;</td>
<td>0.5</td>
</tr>
<tr>
<td>6</td>
<td>(0.0470, 0.0470, 0.0144)</td>
<td>(0.0651, 0.0479, 0.0398)</td>
<td>&lt; 0.0180, 0, 0 &gt;</td>
<td>0.188</td>
</tr>
</tbody>
</table>

Table 3.8 The particle size distribution of each particle insertion region.

<table>
<thead>
<tr>
<th>$d$, μm</th>
<th>1, %</th>
<th>2, %</th>
<th>3, %</th>
<th>4, %</th>
<th>5, %</th>
<th>6, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>274</td>
<td>0.6</td>
<td>2.1</td>
<td>12.4</td>
<td>0.6</td>
<td>2.1</td>
<td>12.4</td>
</tr>
<tr>
<td>381</td>
<td>51.5</td>
<td>68.9</td>
<td>79.0</td>
<td>51.5</td>
<td>68.9</td>
<td>79.0</td>
</tr>
<tr>
<td>471</td>
<td>47.4</td>
<td>29.0</td>
<td>8.6</td>
<td>47.4</td>
<td>29.0</td>
<td>8.6</td>
</tr>
<tr>
<td>610</td>
<td>0.5</td>
<td>-</td>
<td>-</td>
<td>0.5</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
3.4 Results and Discussions

The particle velocity field, average velocity, volume fraction, bed thickness and mass flux contours were compared for different temperatures.

The mass flow rate ($m_{out}$) leaving the inclined plane for the 23 °C model was evaluated and compared to $m_{out}$ from previous experimentation [43]. The model encompassed half of the full width of the incline due to the utilization of the symmetrical boundary, therefore, the mass out of the 23 °C model was multiplied by two for an accurate comparison with the experimental measurements. Once steady state was reached for both the model and the experiment, linear trendlines of the mass out were calculated to compare the mass flow rates, given as:

$$m(t) = \beta_1 t + \beta_0,$$

(3.11)

where $m$ is the mass out; $\beta_1$ is the slope of the linear trendline; and $\beta_0$ is the mass-intercept of the linear trendline. The slopes of the trendlines represent the mass flow rate out of the system at steady state. The $\beta_1$ and $\beta_0$ from the 23 °C model, the experiment, and the other higher temperature models are listed in Table 3.9. A 0.4% difference in steady state mass flow rate was observed between the model and the experiment, indicating the particle insertion accurately captured the experimental flow behavior.

Similarly, $m_{out}$ was calculated and compared at each temperature, as seen in Figure 3.7. Linear trendlines were calculated for respective temperatures after each simulation reached approximately steady-state and corresponding parameters are listed in Table 3.9. The input conditions for each of the models remained the same except for the input flow
properties. The $\dot{m}_{in}$ was non-uniform because the flow properties used were changed for each model. To further verify model fidelity, the $\dot{m}_{in}$ and $\dot{m}_{out}$ of the system were compared. The times at which steady state was reached, $t_s$, and the percent differences between the inlet and outlet mass flow rates for each temperature are listed in Table 3.10. Compared to the 23 °C case, an increase in $\dot{m}_{out}$ was observed for 200 °C and 400 °C, with decreases of 8.7% and 15.6%, respectively. The peak $\dot{m}_{out}$ was observed at 400 °C. A decrease from the peak was observed at 600 °C and 800 °C, where an 8.5% increase in $\dot{m}_{out}$ was observed for 600 °C, and a 37.9% decrease in $\dot{m}_{out}$ was observed for 800 °C. For $t_s$, a decrease was observed for 200 °C and 400 °C, with a 52% and 59% decrease, respectively. The min($t_s$) was also observed at 400 °C. Sharp increases in $t_s$ were then observed from the minimum for 600 and 800 °C, with an overall decrease of 33% at 600 °C and an overall increase of 53% at 800 °C. The changes in both $\dot{m}_{out}$ and $t_s$ corresponded to the changes in $\mu_s$ and $\mu_r$, indicating these parameters had a significant impact on the flow behavior, on $t_s$, and on the particle residence time.

Table 3.9 Parameters determined for linear trendlines of the calculated mass out.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$\beta_1$, g·s$^{-1}$</th>
<th>$\beta_0$, g</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment</td>
<td>12.7</td>
<td>-42.1</td>
</tr>
<tr>
<td>23 °C</td>
<td>6.34</td>
<td>-18.9</td>
</tr>
<tr>
<td>200 °C</td>
<td>6.89</td>
<td>-13.3</td>
</tr>
<tr>
<td>400 °C</td>
<td>7.33</td>
<td>-14.9</td>
</tr>
<tr>
<td>600 °C</td>
<td>6.88</td>
<td>-22.1</td>
</tr>
<tr>
<td>800 °C</td>
<td>3.94</td>
<td>-30.4</td>
</tr>
</tbody>
</table>
Table 3.10 The times when steady state was reached, and the percent difference between the inlet and outlet mass flow rates for each temperature.

<table>
<thead>
<tr>
<th>$T$, °C</th>
<th>$t_\text{s}$, s</th>
<th>$\Delta m \cdot m_{in}^{-1}$, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>10.0</td>
<td>-1.0</td>
</tr>
<tr>
<td>200</td>
<td>4.8</td>
<td>2.3</td>
</tr>
<tr>
<td>400</td>
<td>4.1</td>
<td>1.0</td>
</tr>
<tr>
<td>600</td>
<td>6.7</td>
<td>0.1</td>
</tr>
<tr>
<td>800</td>
<td>15.3</td>
<td>2.1</td>
</tr>
</tbody>
</table>

Figure 3.7 Mass leaving the inclined plane flow geometry as a function of time for all temperatures.

The free-surface velocity magnitude fields and mass flux contours were calculated for each temperature and were averaged over the entire steady state regime in 0.5 s increments. The flow was divided into discrete regions $3.7 \times d_p \times 3.7$ mm$^3$ to calculate the contours, where $d_p = 471$ μm approximates the average particle size. The velocity field contours only included particle layers representative of the free-surface flow, as seen in
Figure 3.8. An increase in the overall surface velocity magnitude was observed as temperature increased to 400 °C. A decrease in surface velocities was then observed as the temperature increased to 800 °C. These changes again corresponded to changes in $\mu_s$ and $\mu_r$, indicative of significant impact on the flow behavior.

The mass flux contours were averaged over the particle bed thickness and calculated as a function of the flow volume fraction and average velocity magnitude, as seen in Figure 3.9. The contours were averaged over the entire steady state regime in 0.5 s increments. The mass flux was calculated as:

$$\dot{m}'' = \varphi \rho V_m,$$

(3.12)

where $\rho$ is the particle density of the Carbobead CP particles previously reported in literature [13]; $\overline{V}_m$ is the average velocity magnitude; and $\varphi$ is the volume fraction, calculated as:

$$\varphi = \frac{4 \pi \sum_{i=1}^{N} r_i^3}{V},$$

(3.13)

where $r$ is the particle radius, $i$ denotes the particle number within the region, $N$ is the total number of particles within the region, and $V$ is the region volume. The $\overline{V}_m$ was determined by averaging the velocity magnitudes of all the particles within a region across the entire particle bed thickness. This resulted in the higher velocity flows having a more centralized $\dot{m}''$ and $V_m$, compared to the lower velocity cases.
Figure 3.8 Velocity magnitude fields of particle flows along an inclined plane at steady state with resolved polydisperse spheres for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.
Figure 3.9 Mass flux contours of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.

An increase in $\dot{m}''$ was observed for higher flow velocity conditions while a decrease in $\dot{m}''$ was observed for temperatures with lower flow velocity. The higher velocity flows displayed more centralized $\dot{m}''$ and $V_m$, indicating smaller shear forces incident on particles, while the lower velocity cases displayed wider low-velocity zones close to the alumina side wall, due to higher shear forces. The changes in $\dot{m}''$ also corresponded to changes in $\mu_s$ and $\mu_t$, again indicative of a significant impact on the flow behavior. The effects of the higher
frictional forces in the 800 °C model resulted in a significant delay in the flow development. A significantly lower mass flux was observed for the 800 °C due to significantly higher static friction coefficients. The effects of the shear force from the alumina side wall in the 800 °C model also resulted in a wide low-velocity region that developed earlier in the flow (Figure 3.8e).

Contours of the flow volume fraction, average velocity magnitude, and particle bed thickness were also developed, as seen in Figures 3.10 - 3.12. Similar flow regions were used in developing the contours, however the parameters were averaged over the entire particle bed thickness. The particle bed thickness contours were represented with a non-dimensional particle bed thickness, \( \tilde{h} \), referenced from the inclined planed surface, calculated as:

\[
\tilde{h} = \frac{h_r}{h_m},
\]

(3.14)

where \( h_r \) is the maximum particle height within the region, and \( h_m \) is the maximum bed height for the model. The values for \( h_m \) were 3.40 mm, 3.39 mm, 3.42 mm, 3.43 mm, and 3.44 mm for 23 °C, 200 °C, 400 °C, 600 °C, and 800 °C, respectively.

Decreases in \( \phi \) and \( \tilde{h} \) were observed for higher velocity flows. Similarly, increases of \( \phi \) and \( \tilde{h} \) were observed for lower velocity flows. The maximum \( \phi \), \( \tilde{h} \), and \( \bar{V}_m \) for each temperature were observed at the symmetrical boundary condition (bottom), while the minimum \( \phi \), \( \tilde{h} \), and \( \bar{V}_m \) was observed at the alumina wall boundary (top), indicating a symmetrical flow was captured. As the length along the incline, \( L \), increased, increases in \( \bar{V}_m \) and decreases in \( \phi \)
and $\tilde{h}$ were observed, indicating gravitational forces on the inclined flow were also captured.

Figure 3.10 Volume fraction contours of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.
Figure 3.11 Average velocity magnitude contours of particle flows along an inclined plane at steady state for (a) 23°C, (b) 200°C, (c) 400°C, (d) 600°C, and (e) 800°C.
Figure 3.12 Non-dimensional particle bed thickness of particle flows along an inclined plane at steady state for (a) 23 °C, (b) 200 °C, (c) 400 °C, (d) 600 °C, and (e) 800 °C.

3.5 Conclusions

Several inclined plane, particle flows of sintered bauxite proppants were examined at temperatures ≤ 800 °C using the DEM. High temperature flow properties necessary to model particle flows were measured for Carbobead CP 30/60. High temperature flow property measurements were adapted from room temperature property measurement
techniques. The particle shape and size distribution of the particles did not change significantly due to thermal expansion. A slight decrease in the elastic modulus and shear modulus was observed at higher temperatures, and a gradual increase in Poisson’s ratio was observed at higher temperatures. The particle-particle coefficient of static sliding friction increased significantly at higher temperatures, while the particle-alumina sliding friction did not have a significant change at higher temperatures. The particle-particle and particle-alumina coefficients of static rolling friction initially decreased at temperatures ≤ 600 °C and then increased as the temperatures approached 800 °C. The particle-particle coefficient of restitution decreased with an increase in temperature, while the particle-alumina restitution coefficient did not change significantly at higher temperatures.

The measured flow properties were used as inputs for numerically modeling the particle flows in LIGGGHTS which employs the discrete element method. The baseline 23 °C model was compared to the room temperature experimental measurement in Chapter 2. Linear trendlines capturing the steady state mass flow rate of the experiment and the model indicated a 0.4% difference in steady state mass flow rate. The masses out for each temperature model were also compared. An 8.7%, 15.6%, and 8.5% increase in steady state mass flow rate was observed for 200 °C, 400 °C, and 600 °C, respectively, while a 37.9% decrease was observed for 800 °C. A 52%, 59%, and 33% decrease in the time to reach steady state was observed for 200 °C, 400 °C, and 600 °C, respectively, while a 53% increase in time was observed for 600 °C and 800 °C. The trends from both the mass flow rate and the time to reach steady state correspond to the observed changes in the static coefficients of friction, indicating these significantly influence the flow behavior and particle resonance time. The free-surface velocity magnitude fields, and the flow volume
fraction, average velocity magnitude, non-dimensional particle bed thickness, and mass flux contours were also compared for each temperature. An overall increase in the free-surface velocity magnitude was observed as temperature increased to 400 °C, and then proceeded to decrease as the temperature approached 800 °C, corresponding to behavior observed for the coefficients of static friction. More centralized mass fluxes and velocity magnitudes were observed for high velocity cases, while low velocity cases displayed wider low-velocity zones close to the alumina side wall, due to higher shear forces. A significant delay in the flow development of the 800 °C model was observed, due to significantly higher values for the static coefficients of friction.
CHAPTER 4. RESEARCH CONTRIBUTIONS AND FUTURE WORK

4.1 Research Contributions

While previous investigations of candidate particulate TES media have sought to characterize bulk granular transport, in this work a comprehensive characterization of the flow behavior of particulate TES media was performed. Room and high temperature characterizations of the flow behavior of Carbobead CP particles, a candidate TES medium, were performed. Measurements of room and high temperature flow properties were completed for particle shape and size distributions, elastic properties, and the coefficient of restitution. A room temperature granular flow experiment of Carbobead CP particles along an inclined plane was completed, and a corresponding room temperature DEM model was developed. Average velocity differences between the experiment and the model were less than 10%.

State-of-the-art high temperature flow models of Carbobead CP particles between 23 °C and 800 °C were developed by optimizing and extending the validated room temperature model. High temperature flow properties were measured and incorporated into the models, and the steady state flow behavior was compared. Validation of the optimized 23 °C model was accomplished by achieving a 0.4% difference from experimental measurement of the steady state mass flow rate. The velocity field, flow volume fraction, average flow velocity, flow bed thickness, and mass flux contours from each high
temperature model were calculated and provided to characterize the high temperature flow behavior.

This work represents a comprehensive effort aimed at addressing the knowledge gap for particulate TES media by accurately determining room and high temperature flow properties and using these properties to develop predictive DEM models of granular flows across a range of temperatures.

4.2 Future Work

The measurement of particle flow properties and the development of particle flow DEM models at several temperatures serve as foundations to accurately predict high temperature particle flows for next-generation SPHRR design. Further study into the coupling of high temperature granular flow models to heat transfer is needed to effectively predict experimentally observed high temperature particle flow behavior. The coupling of the DEM to computational fluid dynamics (CFD) to incorporate heat transfer has been completed previously [40], however models that accurately capture radiative heat transfer relative to CSP need to be further developed and refined. Additionally, the development of a dynamic coupling between granular flow models and comprehensive heat transfer models also needs to be completed as current models are unidirectional, where the measured DEM flow profile is utilized by the CFD software. Finally, the development of coupled DEM granular flow-heat transfer models of several SPHRR geometries could be considered. The current work considers the inclined plane flow configuration, however the investigation of other geometries would greatly contribute to next generation SPHRR design.
APPENDIX A. DATA PROCESSING ALGORITHMS

MATLAB algorithms used to process experimental measurements and model calculations are presented.

A.1 Particle Size and Shape Distributions

A.1.1 Optical Microscopy Image Analysis

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Authors: Andrew Schrader and Justin Yarrington
%%% Date: 9/20/18
% https://imagej.nih.gov/ij/plugins/descriptors.html
% https://en.wikipedia.org/wiki/Shape_factor_(image_analysis_and_microscopy)#Aspect_ratio
% Purpose: The purpose of this code is to extract a particle size distribution from an image of particles
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Set File %
clear all;
close all;
clc;

pname = 'C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y1Q3\Material Testing\Particle Characterization\CarboBead CP 30-60\';
%'I:\CNES\Backup_8_14_17\Research\Fall 2018\Particle Characterization\CarboCast ID50\';
%'I:\CNES\Backup_8_14_17\Research\Fall 2018\Particle Characterization\CarboCast ID50\';
fname = 'Image001_Overlay001';
addpath(pname);
% Load Image %
imgraw = imread(strcat(fname,'.tif'));
imshhow(imgraw);
%title('Raw Image');
%% background?
% bg = imopen(imraw,strel('disk',10));
% figure
% imshow(bg)
% colormap jet
% title Background
%
% Subtract background
% y = imsubtract(imraw,bg);
% figure
% imshow(y)
% title Flattened
%
% Grayscale image
% figure;
% imggray = rgb2gray(y);
% imshow(imggray);
% title('Gray Scale Image');

% %color map
llevel = 0.85;
figure;
A=imbinarize(imraw,'adaptive','Sensitivity',0.6,'ForegroundPolarity','Dark');
%A = im2bw(imraw,graythresh(imraw)*0.85),graythresh(imraw) %0.75
against white background
imshow(A);
% %title('BW image');
% %
%Complement BW image
figure;
A_compliment = imcomplement(A);
imshow(A_compliment); hold on;
A_compliment = imfill(A_compliment,0,8);
% %title('Complement image');
%
% % % %Filled Holes //good for scenarios where background specs are
% prevalent
% % remove all object containing fewer than 30 pixels
% BW2 = bwareaopen(A_compliment,50);
% % % fill a gap in the pen's cap
% se = strel('disk',2);
% % BW2 = imclose(BW2,se);
% % BW2 = imfill(BW2,'holes');
% figure
% imshow(BW2)
% %title('Filled Image')
% %
% % %Attempt to find circles
% %figure;
% % [B,L] = bboundaries(BW2,'noholes');
% % Display the label matrix and draw each boundary
% % new = imshow(label2rgb(L, @jet, [.5 .5 .5]))
%% hold on
%% for k = 1:length(B)
%%    boundary = B{k};
%%    plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2)
%% end
%%
%% [centers, radii] = imfindcircles(BW2, [22 50], 'Sensitivity', 0.5);
%% h = viscircles(centers, radii, 'EdgeColor', 'b');
%%
%% A_compliment = imfill(BW2, 'holes');
over = A_compliment;
figure;
imshow(over);
%title('Overlay');

%% Run Cookie Cutter %%
again = 1;

while (again == 1)
h = imfreehand;
maskit = createMask(h); % Binary mask with 1's inside ROI
over = imoverlay(over, maskit, 'black');
imshow(over)
prompt = 'Another one?';
again = input(prompt)

if (again == 1)

else
    again = 0;
end
end

%% Run Line Cutter %%
again = 1;
while (again == 1)
h = imline;
% set(h, 'LineWidth', 2);
% position = wait(h);
maskit = createMask(h); % Binary mask with 1's inside ROI
over = imoverlay(over, maskit, 'black');
imshow(over)
prompt = 'Another one?';
again = input(prompt)
if (again == 1)

else
    again = 0;
end
end

%% Attempt to find circles %%
figure;
Circle = im2bw(over);
[B, L] = bwboundaries(Circle, 'noholes');

68
% Display the label matrix and draw each boundary
new = imshow(label2rgb(L, @jet, [.5 .5 .5]))
hold on
% for k = 1:size(B)
  % boundary = B(k);
  % plot(boundary(:,2), boundary(:,1), 'k.', 'LineWidth', 2)
  % hold on;
  % bb = minBoundingBox(rot90(boundary));
  % bb_plot = [bb bb(:,1)];
  % plot(bb_plot(1,:),bb_plot(2,:),'-r'); hold on;
  % for i=1:4
  %   dist(i) = sqrt((bb_plot(1,i)-bb_plot(1,i+1))^2+(bb_plot(2,i)-
  %     bb_plot(2,i+1))^2); %pixel dist
  % % end
  %
  % d_feret(k,1:2) = [min(dist) max(dist)];
% end

%% Calibrate pixels %
%length = 592.93; %pixels
scalebar = 250.0; %[microns]
d = imdistline;
pix2dist = scalebar/d;

%% Extract information from circle plot %
stats = regionprops(L, 'Area', 'Centroid');
n_pixels = 168.63;%18.4; %
length = 500.00; %[microns]
pix2dist = length/n_pixels;
threshold = graythresh(imgraw)*1.21;

% Feret's diameter extraction
[ferets,LM] = bwferet(imbinarize(over(:,:,end)),'all');
MaxFerets = ferets.MaxDiameter;
MinFerets = ferets.MinDiameter;
for k=1:size(B)
  max_coord = cell2mat(ferets.MaxCoordinates(k));
  max_coord_a(k,:) = max_coord(1,:);
  max_coord_b(k,:) = max_coord(2,:);
  min_coord = cell2mat(ferets.MinCoordinates(k));
  min_coord_a(k,:) = min_coord(1,:);
  min_coord_b(k,:) = min_coord(2,:);
end

% loop over the boundaries
for k = 1:size(B)

  % obtain (X,Y) boundary coordinates corresponding to label 'k'
  boundary = B{k};
  [a,b] = size(boundary);
  %Iteratively determine the maximum distance between points (Feret's
%diameter
distmax(k) = 0;
for i=1:a
    for j=1:a
        distcheck = sqrt((boundary(i,1) - boundary(j,1))^2 + (boundary(i,2) - boundary(j,2))^2);
        if distcheck>distmax(k)
            distmax(k) = distcheck;
        end
    end
end

% compute a simple estimate of the object's perimeter
delta_sq = diff(boundary).^2;
perimeter(k) = sum(sqrt(sum(delta_sq,2)));%

% obtain the area calculation corresponding to label 'k'
area(k) = stats(k).Area;

% compute the circularity metric
metric(k) = 4*pi*area(k)/perimeter(k).^2;

% compute the roundness metric
round(k) = 4*area(k)/(pi*(MaxFerets(k)).^2);

% Compute estimate of object's diameter
d_eff(k) = (4.0*area(k)/perimeter(k))^*pix2dist;

% Compute maximum and minimum feret diameter of particle
d_feret_min(k) = MinFerets(k).*pix2dist;
d_feret_max(k) = MaxFerets(k).*pix2dist;

%Compute maximum distance between points of a particle
distmax(k) = distmax(k).*pix2dist;

%Compute aspect ratio of current particle
AR(k) = MaxFerets(k)./MinFerets(k);

% display the results
element = sprintf('%d',k);
metric_string = sprintf('%2.2f',metric(k));
d_eff_string = sprintf('%2.2f',d_eff(k));
% d_feret_min = sprintf('%2.2f',d_feret(k,1));
% d_feret_max = sprintf('%2.2f',d_feret(k,2));
d_max = sprintf('%2.2f',distmax(k));
% % mark objects above the threshold with a black circle
% if metric > threshold
%    centroid = stats(k).Centroid;
%    plot(centroid(1),centroid(2),'ko');
% end
Centroids = stats(k).Centroid;

% display the results

% display the results

% display the results
Double check to ensure that ferets and particles are matched

```matlab
m_max = (max_coord_b(k,2)-max_coord_a(k,2))/(max_coord_b(k,1)-
max_coord_a(k,1));

m_min = (min_coord_b(k,2)-min_coord_a(k,2))/(min_coord_b(k,1)-
min_coord_a(k,1));

x_feret_cent(k) = (m_max*max_coord_a(k,1)-
m_min*min_coord_a(k,1))+(min_coord_a(k,2)-max_coord_a(k,2))./(m_max-
m_min);

y_feret_cent(k) = m_max*(x_feret_cent(k)-
max_coord_a(k,1))+max_coord_a(k,2);
```

```matlab
for (kk=1:size(B))
    dist(kk) = sqrt((stats(kk).Centroid(1)-
x_feret_cent(k))^2+(stats(kk).Centroid(2)-y_feret_cent(k))^2);
end

loc = find(dist==min(dist));

if (loc == k)
    % do nothing
else
    disp('WARNING: Feret Diameters not matched with the appropriate
    centroid');
end
```

```matlab
Data(k,:) = [k metric(k) round(k) d_eff(k) distmax(k) d_feret_min(k)
d_feret_max(k) AR(k) area(k) perimeter(k)];
```

```matlab
[l,w] = size(Data);
ccc=1;
ddd=1;
for p=1:l
    if (Data(p,4) ~= 0) && (Data(p,2) < 1)
        Data1(ccc,:) = Data(p,:);
        if Data1(ccc,3) > 100
            DataNew(ddd,:) = Data1(ccc,:);
            ddd=ddd+1;
        end
        ccc=ccc+1;
    end
end

end
```
bw=imbinarize(over,'adaptive');
[out, LM] = bwferet(bw(:,:,end),'all');

save(strcat(pname,'\',fname,'.mat'),'Data');

%% Save Image As is to avoid loss of edited figure %%
export_fig('C:\Users\ajsch\OneDrive\Documents\Post Doc\Fall 2019\Edited.tif',gca);

%% Accidental Close Startup %%
figure;
over = imread('C:\Users\ajsch\OneDrive\Documents\Post Doc\Fall 2019\Edited.tif');
imshow(over);

A.1.2  Size and Shape Distribution Algorithm

clear all;
close all;
clc;

%read in compiled image data
data=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q2\Material Testing\Particle Characterization\Carbobead CP Particle Characterization.xlsx',1,'C2:I109');
deff=data(:,3);
dmax=data(:,6);
dmin=data(:,5);
ar=data(:,7);
circ=data(:,1);
round=data(:,2);

%plot properties

set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');

%%
a = figure('Position',[1 1 3.25 10]);

%histogram(deff,25)

axes('Position',[0.5 7 2.5 2.5]);
[h,hpd]=histfitparam(deff,25,'lognormal');
h(1).FaceColor = [.55 .55 .55];
set (h(2),'Color','black');

72
set (h(2), 'Linewidth', 1.5)
fmt_pubfig('d_{\text{eff}}, \: \mu \text{m}', 'n_{\text{bin}}, -')
axis([300 575 0 12])
title('(a)')

axes('Position', [0.5 3.75 2.5 2.5]);
hmax(1).FaceColor = [.55 .55 .55];
set (hmax(2), 'Color', 'black');
set (hmax(2), 'Linewidth', 1.5)
fmt_pubfig('d_{\text{major}}, \: \mu \text{m}', 'n_{\text{bin}}, -')
axis([375 900 0 15])
title('(b)')

axes('Position', [0.5 0.5 2.5 2.5]);
hmin(1).FaceColor = [.55 .55 .55];
set (hmin(2), 'Color', 'black');
set (hmin(2), 'Linewidth', 1.5)
fmt_pubfig('d_{\text{minor}}, \: \mu \text{m}', 'n_{\text{bin}}, -')
axis([325 625 0 10])
title('(c)')

b = figure('Position', [1 1 3.25 10]);
axes('Position', [0.5 0.5 2.5 2.5]);
har(1).FaceColor = [.55 .55 .55];
set (har(2), 'Color', 'black');
set (har(2), 'Linewidth', 1.5)
fmt_pubfig('AR, \: \text{-}', 'n_{\text{bin}}, -')
axis([1 1.8 0 22])
title('(c)')

axes('Position', [0.5 7 2.5 2.5]);
hcirc(1).FaceColor = [.55 .55 .55];
set (hcirc(2), 'Color', 'black');
set (hcirc(2), 'Linewidth', 1.5)
fmt_pubfig('C, \: \text{-}', 'n_{\text{bin}}, -')
axis([0.4 0.85 0 20])
title('(a)')

axes('Position', [0.5 3.75 2.5 2.5]);
hr(1).FaceColor = [.55 .55 .55];
set (hr(2), 'Color', 'black');
%set (hr(2), 'Linestyle', '--')
A.2 Elastic Properties

clear all
close all
clc

set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');
data=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\Material Testing\Elastic Properties\High Temperature CP.xlsx',1,'H2:J80');
temp=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\Material Testing\Elastic Properties\High Temperature CP.xlsx',1,'A2:A80');
elastic=data(:,1);
shear=data(:,2);
poisson=data(:,3);
a = figure('Position',[1 1 3.5 6.5]);
%bottom
axes('Position',[0.55 0.5 2.5 2.5]);
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',10);
plot(temp,poisson,'Color','k');
fmt_pubfig('T \(\text{,} \) \(\circ \text{C}\)', '
u \(\text{,} \) -')
axis([23 800 0.2 0.3])
title('{(b)}')

%middle
axes('Position',[0.55 3.5 2.5 2.5]);
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',10);
plot(temp,shear,'Color','k');
hold on
text(650,85,'G')
plot(temp,elastic,'Color','k');
text(650,205,'E')
fmt_pubfig('T \(\text{,} \) \(\circ \text{C}\)', '\(\text{GPa}\)')
A.3 Coefficient of Restitution

A.3.1 Image Processing and PTV Algorithm

%% Authors: Andrew Schrader and Justin Yarrington
%% Date: 2/20/19

%% Set File %
clear all;
close all;
clc;

name_array = ['A';'B'; 'C';'D';'E';'F'];
frame_rate = 2000; %[f/sec]
n_frames_per_analysis = 5;
dt = n_frames_per_analysis/frame_rate; %[s]

%% Calibrate Pixel %
figure(1);
fname = name_array(1,:);
% [imgraw,map] = imread((strcat(fname,'.tif')));
imshow(imgraw,map);
d = imdistline
prompt = 'mm/pixels ?: '; %pix2dist = input(prompt); %[microns/pixels] sensitivity and calibration of image

%above commented out if determined in prior image
pix2dist = 3.33/34;
%% Determine normal vector to impact plane %%

% figure(1);
% fname = name_array(1,:);
% [imgraw,map] = imread((strcat(fname,'.tif')));
% imshow(imgraw,map);
% % Perform Edge Extraction
% % edge detection
% [imedge,threshOut] = edge(im2double(imgraw),'Canny',0.1,1.0);
% figure;
% imshow(imedge);
% % select two points for the stem (assuming plane is normal to the stem)
% % enter them into the following
% %
% prompt = 'point 1 x upper ?: '
% x_1 = input(prompt);
% prompt = 'point 1 y upper ?: '
% y_1 = input(prompt);
% prompt = 'point 2 x lower ?: '
% x_2 = input(prompt);
% prompt = 'point 2 y lower ?: '
% y_2 = input(prompt);
% %
% n = (x_1-x_2, y_1-y_2)./sqrt((x_1-x_2).^2+(y_1-y_2).^2); % vector normal to impact plane
%
% above commeted out if determined in prior images
n = [0.0 -1.0];
n_mag = norm(n);

%% Load Image and perform PTV %%
for i=1:6
fname = name_array(i,:); 

figure(i);
[imgraw,map] = imread((strcat(fname,'.tif')));
imnew=imcrop([imgraw,map],[75,185,360-75,357-185]);
imshow(imgraw,map);

% Perform Edge Extraction
% edge detection
[imedge,threshOut] = edge(im2double(imgraw),'Canny',0.1,1.0);% figure;
imshow(imedge);

% Run manual particle location%
figure(i);
again = 1;
count = 1;
while (again==1)
h = imellipse;
pause;
pos(count,:) = getPosition(h);

end

end
again = 0;
count = count+1;
if (again == 1)
else
again = 0;
end
end

%Run ellipse fitter
figure(i);
[r,c] = size(pos);
Ellipse_y_data_raw = zeros(1,1000);
Ellipse_x_data_raw = zeros(1,1000);
e_test = imellipse(gca, pos(1,:));

%Mask creation
maskit= createMask(e_test); % Binary mask with 1's inside ROI
over = imgedge;
over(~maskit) = 0; % blacken outside of the mask
[y,x] = find(over > 0.9);
if length(x) == 0
[xo_in,yo_in,P] = impixel;
else
hold on;
plot(x,y,'.g'); hold on;
ellipse_t = fit_ellipse(x,y);
a = ellipse_t.a;
b = ellipse_t.b;
alpha = ellipse_t.phi;
xo = ellipse_t.X0;
yo = ellipse_t.Y0;
xo_in = ellipse_t.X0_in;
yo_in = ellipse_t.Y0_in;
long_axis = ellipse_t.long_axis;
short_axis = ellipse_t.short_axis;

R = [cos(alpha) sin(alpha); -sin(alpha) cos(alpha)];

% the ellipse
theta_r = linspace(0,2*pi);
ellipse_x_r = xo + a*cos( theta_r );
ellipse_y_r = yo + b*sin( theta_r );
rotated_ellipse = R * [ellipse_x_r;ellipse_y_r];

hold on;
plot(rotated_ellipse(1,:),rotated_ellipse(2,:),'g'); hold on;

%Plot the centroid number
element = sprintf('%d',i);

text(xo_in,yo_in,element,'Color','r','FontSize',12,'FontWeight','bold');
plot(xo_in,yo_in,'.r');
end
Ellipse_Data(i,:) = [i xo_in yo_in];
end

%Extract displacement of particle
d(1) = sqrt((Ellipse_Data(2,2) - Ellipse_Data(1,2))^2+(Ellipse_Data(2,3)-Ellipse_Data(1,3))^2); %/pixels
d(2) = sqrt((Ellipse_Data(3,2) - Ellipse_Data(2,2))^2+(Ellipse_Data(3,3)-Ellipse_Data(2,3))^2);
d(3) = sqrt((Ellipse_Data(5,2) - Ellipse_Data(4,2))^2+(Ellipse_Data(5,3)-Ellipse_Data(4,3))^2);
d(4) = sqrt((Ellipse_Data(6,2) - Ellipse_Data(5,2))^2+(Ellipse_Data(6,3)-Ellipse_Data(5,3))^2);

%Extract unit, particle path direction vector
u(1,:) = [(Ellipse_Data(2,2) - Ellipse_Data(1,2), Ellipse_Data(2,3)-Ellipse_Data(1,3))./d(1);
u(2,:) = [(Ellipse_Data(3,2) - Ellipse_Data(2,2), Ellipse_Data(3,3)-Ellipse_Data(2,3))./d(2);
u(3,:) = [(Ellipse_Data(5,2) - Ellipse_Data(4,2), Ellipse_Data(5,3)-Ellipse_Data(4,3))./d(3);
u(4,:) = [(Ellipse_Data(6,2) - Ellipse_Data(5,2), Ellipse_Data(6,3)-Ellipse_Data(5,3))./d(4);

for i=1:4
v(i) = (pix2dist./1000.0).*d(i).*dot(u(i,:),n)./dt; %[m/s] velocity
mag_normal = norm(u(i,:));
v_mag(i) = (pix2dist./1000).*d(i)./dt; %[m/s], total velocity magnitude

end
v_in = abs(mean(v_mag(1:2))); %[m/s], average in
v_out = abs(mean(v_mag(3:4))); %average out

theta_in = acos(abs(dot(u(2,:),n))/(mag_normal*n_mag)); %[degrees]
theta_out = acos(abs(dot(u(3,:),n))/(mag_normal*n_mag)); %[degrees]

COR = abs(mean(v(3:4)))./mean(v(1:2))
COR_max = abs(v(3)./v(2))

%Save Data
save('Test.mat','Ellipse_Data','d','v','COR','COR_max','pix2dist','v_mag','v_in','v_out','theta_in','theta_out');
A.3.2 Post-processing Algorithms

A.3.2.1 Convection and Radiation Heat Loss

```matlab
clear all
close all
clc

Tpdrop=629.844;  %Input measurement from COR test
Timpact=592.318; %Input measurement from COR test
Tair=343.344;    %Input measurement from COR test
dt=10/1000;     %10ms
Dheat=3.5*25.4/1000; %Effective diameter of the heater
l=2.25*25.4/1000; %length between heater and particle
nuair=76.37*10^(-6); %Dependent on input temperature
prair=0.702;       %Dependent on input temperature
kair=54.9*10^(-3); %Dependent on input temperature
Cp=1100;           %Dependent on input temperature

Vparticles=[0.8183, 0.9610, 1.1033, 1.0193, 1.0338, 1.0193, 1.082, 1.0411];
Vave=mean(Vparticles);
Dparticles=425.2183*10^(-6);
mass=4/3*pi*(Dparticles/2)^3*3270;
As=4*pi*(Dparticles/2)^2;

Aheat=0.00258064;

Re=Vave*Dparticles/nuair;
Nu=2+0.6*Re^(1/2)*prair^(1/3);
h=Nu*kair/Dparticles;

R1=Dheat/(2*l);
R2=Dparticles/(2*l);
S=1+((1+R2^2)/(R1^2));
Fheatpart=0.5*(S-(S^2-(4*(Dparticles/Dheat)^2))^2)^(1/2));

%Energy balance
qrad=Aheat*Fheatpart*5.67*10^(-8)*(1673.15^4-512.15^4);
qconv=h*As*(Tpdrop-Tair);
Timpact=(qrad+qconv)*dt/(mass*Cp)+Tair
```

A.3.2.2 Statistical Parameters and figure

```matlab
clear all
close all
clc

%input data
```
data=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q2\Material Testing\High Temp COR\processcor.xlsx',1,'A1:J10'); %cp-alumina
data2=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q2\Material Testing\High Temp COR\processcor.xlsx',2,'A1:J10'); %cp-cp
p=1;
while p<3
    room(:,p)=data(:,p);
t200c(:,p)=data(:,p+2);
t400c(:,p)=data(:,p+4);
t600c(:,p)=data(:,p+6);
t800c(:,p)=data(:,p+8);
    room2(:,p)=data2(:,p);
t200c2(:,p)=data2(:,p+2);
t400c2(:,p)=data2(:,p+4);
t600c2(:,p)=data2(:,p+6);
t800c2(:,p)=data2(:,p+8);
    p=p+1;
end
room2(:,1)=22*ones(length(room2(:,1)),1);
t200c2(:,1)=200*ones(length(t200c2(:,1)),1);
t400c2(:,1)=400*ones(length(t400c2(:,1)),1);
t600c2(:,1)=600*ones(length(t600c2(:,1)),1);
t800c2(:,1)=800*ones(length(t800c2(:,1)),1);
room(:,1)=22*ones(length(room2(:,1)),1);
t200c(:,1)=200*ones(length(t200c2(:,1)),1);
t400c(:,1)=400*ones(length(t400c2(:,1)),1);
t600c(:,1)=600*ones(length(t600c2(:,1)),1);
t800c(:,1)=800*ones(length(t800c2(:,1)),1);
set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');
a=figure('Position',[1 1 3.5 3.5]);
axes('Position',[0.5 0.5 2.5 2.5]);
testvar=[room; t200c; t400c; t600c; t800c]; %cp-cp
testvar2=[room2; t200c2; t400c2; t600c2; t800c2]; %cp-alumina
boxplot(testvar2(:,2),testvar2(:,1),'Colors','k','Symbol','+k')
ylim([0 1])
fmt_pubfig('T\text{} \: \epsilon \text{} -')
print(a,'-dtiff','-r600','corhightemp')

A.4 Room Temperature Model Post-processing

clear all
close all

80
clc

modelfile='C:\Users\jyarrington3\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q1\LIGGGHTS\processflowdata\piv-model_compare.xlsx';
pivfile='C:\Users\jyarrington3\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q1\LIGGGHTS\processflowdata\piv-model_compare.xlsx';
modelldatabefore(:,:,1)=xlsread(modelfile,4,'V2:X3541'); % model @ t-dt
% the following line corresponds to the "CDT" data (variable modelldata)
modelldata(:,:,1)=xlsread(modelfile,12,'V2:X3387'); %model @ t %vmag ynorm xnorm
modelldataafter(:,:,1)=xlsread(modelfile,9,'V2:X3387'); % model @ t+dt
% vmag ynorm xnorm
% the following line corresponds to the EPSD2 data (variable modelepsd)
modelepsd(:,:,1)=xlsread(modelfile,13,'V2:X821'); %epsd model @ t
% the following line corresponds to the measured data (variable pivdata).
pivdata(:,:,1)=xlsread(pivfile,1,'L2:N386'); % experiment @ t. vmag ynorm xnorm
pivdatab(:,:,1)=xlsread(pivfile,1,'L2:N386'); % experiment @ t-dt.
pivdataa(:,:,1)=xlsread(pivfile,3,'L2:N386'); % experiment @ t+dt.

stepsize=pivdata(11,2)-pivdata(10,2);
cellmax=length(pivdata(:,:,1));
pivmean=mean(pivdata(:,:,1));
modelmean=mean(modelldata(:,:,1));
pivsd=std(pivdata(:,:,1));
modelsd=std(modelldata(:,:,1));

% for i = cellmax:-1:1
% if (pivdata(i,1) > pivmean+3*pivsd) || (pivdata(i,1) < pivmean-3*pivsd)
%    pivdata(i,:)=[];
% elseif (modelldata(i,1) > modelmean+3*modelsd) || (modelldata(i,1)
% < modelmean-3*modelsd)
%    modelldata(i,:)=[];
% end
% end

cellmax=length(pivdata(:,:,1));
px=5;
py=cellmax/px;
ml=length(modelldata(:,:,1));
mla=length(modelldataafter(:,:,1));
mlb=length(modelldatabefore(:,:,1));
temp=stepsize/8;
lepsd=length(modelepsd(:,1));
ysection=5; %section length in mm

%% average to match piv cell size | 1-to-1 comparison
k=0;
p=0;
cellpiv=pivdata(:,1);
for i = 1:1:cellsmax
    zonemaxy=stepsize*(k+1);
    zoneminy=stepsize*(k-1);
    zoneminx=stepsize*(p-1);
    zonemaxx=stepsize*(p+1);
    cellave=0;
count=0;
countb=0;
cellavebefore=0;
cellaveafter=0;
counta=0;
    for j=1:1:ml
        if (modeldata(j,2)>=zoneminy) && (modeldata(j,2) < zonemaxy) &&
            (modeldata(j,3)>=zoneminx) && (modeldata(j,3) < zonemaxx)
            cellave=cellave+modeldata(j,1);
        end
    end
    for j=1:1:mlb
        if (modeldatabefore(j,2)>=zoneminy) && (modeldatabefore(j,2) <
            zonemaxy) && (modeldatabefore(j,3)>=zoneminx) && (modeldatabefore(j,3) <
            zonemaxx)
            cellavebefore=cellavebefore+modeldatabefore(j,1);
        end
    end
    for j=1:1:mla
        if (modeldataafter(j,2)>=zoneminy) && (modeldataafter(j,2) <
            zonemaxy) && (modeldataafter(j,3)>=zoneminx) && (modeldataafter(j,3) <
            zonemaxx)
            cellaveafter=cellaveafter+modeldataafter(j,1);
        end
    end
    cellmodelave(i,1)=(cellmodel(i,1));%+cellmodelb(i,1)+cellmodela(i,1))/3
    cellmodel(i,1)=cellave/count;
    cellmodelb(i,1)=cellavebefore/countb;
    cellmodela(i,1)=cellaveafter/counta;
    cellmodelave(i,1)=(cellmodel(i,1));%+cellmodelb(i,1)+cellmodela(i,1))/3
    p=p+1;
    if mod(i,px)==0
        p=0;
        k=k+1;
    end
end

%% average along x-direction | all y locations
for i = 1:1:py
    zonemax=stepsize*(i+1);
    zonemin=stepsize*(i-1);
    ytempmin=stepsize*i-temp;
    ytempmax=stepsize*i+temp;
    yave=0;
    count=0;
    countb=0;
    counta=0;
    yaveb=0;
    yavea=0;
    countpiv=0;
    ypivave=0;
    ypivaveb=0;
    countpivb=0;
    ypivavea=0;
    countpiva=0;
    yepsdave=0;
    counte=0;
    for j=1:1:ml
        if (modeldata(j,2)>=zonemin) && (modeldata(j,2) < zonemax)
            yave=yave+modeldata(j,1);
            count=count+1;
        end
    end
    for j=1:1:lepsd
        if (modelepsd(j,2)>=zonemin) && (modelepsd(j,2) < zonemax)
            yepsdave=yepsdave+modelepsd(j,1);
            counte=counte+1;
        end
    end
    for j=1:1:mlb
        if (modeldatabefore(j,2)>=zonemin) && (modeldatabefore(j,2) < zonemax)
            yaveb=yaveb+modeldatabefore(j,1);
            countb=countb+1;
        end
    end
    for j=1:1:mla
        if (modeldataafter(j,2)>=zonemin) && (modeldataafter(j,2) < zonemax)
            yavea=yavea+modeldataafter(j,1);
            counta=counta+1;
        end
    end
    for j=1:1:cellsmax
        if (pivdata(j,2)> ytempmin) && (pivdata(j,2) < ytempmax)
            ypivave=ypivave+pivdata(j,1);
            countpiv=countpiv+1;
        end
        if (pivdatab(j,2)> ytempmin) && (pivdatab(j,2) < ytempmax)
            ypivaveb=ypivaveb+pivdatab(j,1);
            countpivb=countpivb+1;
        end
    end
end
if (pivdataa(j,2) > ytempmin) && (pivdataa(j,2) < ytempmax)
    ypivavea=ypivavea+pivdataa(j,1);
    countpiva=countpiva+1;
end

div
end

ymodel(i,1)=yave/count;
ymodelb(i,1)=yaveb/countb;
ymodela(i,1)=yavea/counta;
% ymodel(i,1)=(ymodel(i,1)+ymodelb(i,1)+ymodela(i,1))/3;
ypiv(i,1)=ypivave/countpiv;
ypivb(i,1)=ypivaveb/countpivb;
ypiva(i,1)=ypivavea/countpiva;
yepsd(i,1)=yepsdave/counte;

div
%

%% average along y-direction | all x locations
for i = 1:1:px
    zonemax=stepsize*(i+1);
    zonemin=stepsize*(i-1);
    xtempmin=stepsize*i-temp;
    xtempmax=stepsize*i+temp;
    xave=0;
    xaveb=0;
    xavea=0;
    counta=0;
    countb=0;
    count=0;
    countpiv=0;
    xpivave=0;
    for j=1:1:ml
        if (modeldata(j,3) >= zonemin) && (modeldata(j,3) < zonemax)
            xave=xave+modeldata(j,1);
            count=count+1;
        end
    end
end
for j=1:1:mlb
    if (modeldatabefore(j,3) >= zonemin) && (modeldatabefore(j,3) < zonemax)
        xaveb=xaveb+modeldatabefore(j,1);
        countb=countb+1;
    end
end
for j=1:1:mla
    if (modeldataafter(j,3) >= zonemin) && (modeldataafter(j,3) < zonemax)
        xavea=xavea+modeldataafter(j,1);
        counta=counta+1;
    end
end
xmodelb(i,1)=xaveb/countb;
xmodela(i,1)=xavea/counta;

for j=1:cellsmax
    if (pivdata(j,3)> xtempmin) && (pivdata(j,3) < xtempmax)
        xpivave=xpivave+pivdata(j,1);
        countpiv=countpiv+1;
    end
end

xmodel(i,1)=xave/count;
% xmodel(i,1)=(xmodel(i,1)+xmodelb(i,1)+xmodela(i,1))/3;
xpiv(i,1)=xpivave/countpiv;
end

%% average over the x direction | section the y direction
cellylength=max(pivdata(:,2))*1000/py;
mergenum=ceil(ysection/cellylength);
realylength=max(pivdata(:,2))*1000/mergenum;
countm=0;
mergey=0;
mergeypiv=0;
mergeypiva=0;
mergeypivb=0;
mergeyepsd=0;
q=1;
for i = 1:py
    if mod(i,mergenum)==0
        mergey=mergey+ymodel(i,1);
        countm=countm+1;
        mergeypiv=mergeypiv+ypiv(i,1);
        mergeypiva=mergeypiva+ypiva(i,1);
        mergeypivb=mergeypivb+ypivb(i,1);
        mergeyepsd=mergeyepsd+yepsd(i,1);
        mergemodel(q,1)=mergey/countm;
        mergepiv(q,1)=mergeypiv/countm;
        mergepiva(q,1)=mergeypiva/countm;
        mergepivb(q,1)=mergeypivb/countm;
        mergeepsd(q,1)=mergeyepsd/countm;
        mergey=0;
        mergeypiv=0;
        mergeypiva=0;
        mergeypivb=0;
        mergeyepsd=0;
        countm=0;
        q=q+1;
    elseif i==py
        mergey=mergey+ymodel(i,1);
        mergeypiv=mergeypiv+ypiv(i,1);
        mergeypiva=mergeypiva+ypiva(i,1);
        mergeypivb=mergeypivb+ypivb(i,1);
        mergeyepsd=mergeyepsd+yepsd(i,1);
        countm=countm+1;
        mergemodel(q,1)=mergey/countm;
        mergepiv(q,1)=mergeypiv/countm;
mergepiva(q,1)=mergeypiva/countm;
mergepivb(q,1)=mergeypivb/countm;
mergeepsd(q,1)=mergeyepsd/countm;

else
    mergey=mergey+ymodel(i,1);
    mergeypiv=mergeypiv+ypiv(i,1);
    mergeypiva=mergeypiva+ypiva(i,1);
    mergeypivb=mergeypivb+ypivb(i,1);
    mergeyepsd=mergeyepsd+yepsd(i,1);
    countm=countm+1;
end

%% Calculate pearson coefficients
ymodave=mean(ymodel(:,1));
ypivave=mean(ypiv(:,1));
xmodave=mean(xmodel(:,1));
xpivave=mean(xpiv(:,1));
cellmodave=mean(cellmodelave(:,1));
cellpivave=mean(cellpiv(:,1));
lmerge=length(mergedmodel(:,1));
mergepivave=mean(mergepiv(:,1));

for i = 1:1:py
    ydx(i,1)=(ymodel(i,1)-ymodave).*ypiv(i,1);
    dy2(i,1)=(ypiv(i,1)-ypivave).^2;
    dx2(i,1)=(ymodel(i,1)-ymodave).^2;
end
ydx=sum(ydx(:,1));
dy2=sum(dy2(:,1));
dx2=sum(dx2(:,1));
ry=ydx/(dy2*dx2)^(1/2);

for i=1:1:px
    ydx(i,1)=(xmodel(i,1)-xmodave).*xpiv(i,1);
    dy2(i,1)=(xpiv(i,1)-xpivave).^2;
    dx2(i,1)=(xmodel(i,1)-xmodave).^2;
end
ydx=sum(ydx(:,1));
dy2=sum(dy2(:,1));
dx2=sum(dx2(:,1));
rx=ydx/(dy2*dx2)^(1/2);

for i=1:1:cellsmax
    ydx(i,1)=(cellmodelave(i,1)-cellmodave).*cellpiv(i,1);
    dy2(i,1)=(cellpiv(i,1)-cellpivave).^2;
    dx2(i,1)=(cellmodelave(i,1)-cellmodave).^2;
end
ydx=sum(ydx(:,1));

for i=1:1:cellsmax
    ydx(i,1)=(cellmodelave(i,1)-cellmodave).*cellpiv(i,1);
    dy2(i,1)=(cellpiv(i,1)-cellpivave).^2;
    dx2(i,1)=(cellmodelave(i,1)-cellmodave).^2;
end
ydx=sum(ydx(:,1));
dy2 = sum(dy2(:,1));
dx2 = sum(dx2(:,1));

rcell = ydx / (dy2 * dx2)^(1/2);

for i = 1:1:lmerge
    ydx(i,1) = (mergemodel(i,1) - mergemodave) .* mergepiv(i,1);
dy2(i,1) = (mergepiv(i,1) - mergepivave)^2;
dx2(i,1) = (mergemodel(i,1) - mergemodave)^2;
end
ydx = sum(ydx(:,1));
dy2 = sum(dy2(:,1));
dx2 = sum(dx2(:,1));

rmerge = ydx / (dy2 * dx2)^(1/2);

x = 0:0.001:0.3;

%% Least Squares
% p = length(mergepiv(:,1));
% ymean = mean(mergepiv(:,1));
% xmean = mean(mergemodel(:,1));
% ysum = sum(mergepiv(:,1));
% xsum = sum(mergemodel(:,1));
% xysum = sum(mergepiv.*mergemodel);
% x2sum = sum(mergemodel(:,1)^2);
% b1 = ((xysum) - ((ysum * xsum) / p)) / ((x2sum) - ((xsum)^2) / p);
% b0 = ymean - b1 * xmean;
% residuals = mergepiv - (b0 + b1 * mergemodel);
% SSe = sum(residuals.^2);
% var = SSe / (p - 2);
% st = (max(mergemodel(1,:)) -
%      min(mergemodel(1,:))) / (length(mergemodel(1,:)) - 1);
% xstep = st; 
% yint = (b1 * x + b0)';
% xreal = 182.5:5:277.5;
% pubfig
% plot(x, y, 'k')
% hold on
% plot(x, yint, '--k')
% hold on

%% Least Squares for low friction
ymodel = ymodela;
cellylength = max(pivdata(:,2)) * 1000 / py;
mergenum = ceil(ysection/cellylength);
reallength = max(pivdata(:,2)) * 1000 / mergenum;
countm = 0;
mergey = 0;
mergeypiv = 0;
q=1;
for i = 1:1:py
    if mod(i, mergenum)==0
        mergey=mergey+ymodel(i,1);
        countm=countm+1;
        mergeypiv=mergeypiv+ypiv(i,1);
        mergemodel2([q,1])=mergey/countm;
        mergepiv([q,1])=mergeypiv/countm;
        mergey=0;
        mergeypiv=0;
        countm=0;
        q=q+1;
    elseif i==py
        mergey=mergey+ymodel(i,1);
        mergeypiv=mergeypiv+ypiv(i,1);
        countm=countm+1;
        mergemodel2([q,1])=mergey/countm;
        mergepiv([q,1])=mergeypiv/countm;
    else
        mergey=mergey+ymodel(i,1);
        mergeypiv=mergeypiv+ypiv(i,1);
        countm=countm+1;
    end
end
% p=length(mergepiv(:,1));
% ymean=mean(mergepiv(:,1));
% xmean=mean(mergemodel2(:,1));
% ysum=sum(mergepiv(:,1));
% xsum=sum(mergemodel2(:,1));
% xysum=sum(mergepiv.*mergemodel2);
% x2sum=sum(mergemodel2(:,1).^2);
% b1=((xysum)-((ysum*xsum)/p))/((x2sum)-((xsum)^2)/p));
% b0=ymean-b1*xmean;
% residuals=mergepiv-(b0+b1.*mergemodel2);
% SSE=sum(residuals.^2);
% var=SSE/(p-2);
% stlow=(max(mergemodel2(:,1))-min(mergemodel2(:,1)))/length(mergemodel2(:,1));
% xsteplow=min(mergemodel2(:,1)):stlow:max(mergemodel2(:,1));
% yintlow=(b1*(x)+b0)';
% plot(x,yintlow,'-.k')
% fmt_pubfig('V_{\mathrm{m}}, \: \mathrm{m \cdot s^{-1}} ','V_{\mathrm{e}}, \: \mathrm{m \cdot s^{-1}}')
% legend('\it{I}','\mu_{\mathrm{RF}}','(\mu - \sigma)_{\mathrm{RF}}')
% legend('boxoff')

% pubfig
% barlow=zeros(20,1);
% barlow=abs(0-mergemodel(:,1));
% barhigh=mergemodel2-mergemodel;
% errorbar(xreal,mergemodel(:,1),barlow(:,1),barhigh(:,1),'--
% sk','LineWidth',0.25,'MarkerSize',5);
% hold on
% plot(xreal,mergeepsd(:,1),'-.dk')
% hold on
% plot(xreal,mergepiv(:,1),'-ok','LineWidth',1,'MarkerSize',5)
% fmt_pubfig('L_{\rm{p}}, \: \rm{mm} ','V, \: \rm{m \cdot s^{-1}}')
% legend('CDT Model','EPSD Model','Measured')
% legend('boxoff')
% axis([180 280 0 0.35])

pubfig
plot(xreal,mergemodel(:,1),'--sk','MarkerFaceColor',[0.7 0.7 0.7])
hold on
plot(xreal,mergeepsd(:,1),'-.dk')
hold on
er=ones(20,1)*0.0215;
errorbar(xreal,mergepiv(:,1),er(:,1),er(:,1),'-ok')
%plot(xreal,mergepiv(:,1),'-ok')
fmt_pubfig('L_{\rm{p}}, \: \rm{mm} ','U, \: \rm{m \cdot s^{-1}}')
legend('CDT Model','EPSD2 Model','Measured')
legend('boxoff')
axis([180 280 0 0.35])

A.4.1 Sub-function: Pubfig

function pubfig

set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');

f = figure('OuterPosition',[5 5 5 4],'Position',[12 4 3.3
2.8],'PaperPosition',[0 0 3 3], ...
   'PaperSize',[2.5 2.5],'PaperPositionMode','manual');
axes('Position',[0.65 0.5 2 2]);

A.4.2 Sub-function: Fmt_pubfig

function fmt_pubfig(varargin)

if nargin == 2
    ylabel(['$$$\it{'} varargin{2} ' $$$'], ...
       'Fontsize',11,'Fontname','Times New
Roman','interpreter','latex');
    xlabel(['$$$\it{'} varargin{1} ' $$$'], ...
       'Fontsize',11,'Fontname','Times New
Roman','interpreter','latex');
end

box on;
set(gca,'XMinorTick','on');
set(gca,'YMinorTick','on');
A.5 High Temperature Model Post-processing

A.5.1 Mass Out Algorithm

```matlab
%% read data
close all
clear all
clc

% read in modeled data
dataroom=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\regionroom\room_massout.xlsx',1,'A2:I999999');
data200=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\c200\c200_massout.xlsx',1,'A2:I999999');
data400=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\c400\c400_massout.xlsx',1,'A2:I999999');
data600=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\c600\c600_massout.xlsx',1,'A2:I999999');
data800=xlsread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\c800\c800_massout.xlsx',1,'A2:I999999');

%% plot data
load('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\Library\Paper 2\Contour Plots\roommassflow-experiment.mat')
set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');

dt=0.01;
time=0:dt:22.07;
ltime=length(time);
[l1room, wroom]=size(dataroom);
[ldata, wdata]=size(data200);
[l400, w400]=size(data400);
[l600, w600]=size(data600);
[l800, w800]=size(data800);
massroom=0;
masstotal=0;
mass400=0;
mass600=0;
mass800=0;
```

for i=1:1:ltime-1
    for p=1:1:lroom
        if (dataroom(p,2)>=time(i)) && (dataroom(p,2)<time(i+1))
            massroom=massroom+(4/3*pi*(dataroom(p,3)/2)^3)*3270;
        end
    end
    for j=1:1:ldata
        if (data200(j,2)>=time(i)) && (data200(j,2)<time(i+1))
            masstotal=masstotal+(4/3*pi*(data200(j,3)/2)^3)*3270;
        end
    end
    for k=1:1:400
        if (data400(k,2)>=time(i)) && (data400(k,2)<time(i+1))
            mass400=mass400+(4/3*pi*(data400(k,3)/2)^3)*3270;
        end
    end
    for m=1:1:l600
        if (data600(m,2)>=time(i)) && (data600(m,2)<time(i+1))
            mass600=mass600+(4/3*pi*(data600(m,3)/2)^3)*3270;
        end
    end
    for n=1:1:l800
        if (data800(n,2)>=time(i)) && (data800(n,2)<time(i+1))
            mass800=mass800+(4/3*pi*(data800(n,3)/2)^3)*3270;
        end
    end
    massoutroom(i)=massroom;
    massout(i)=masstotal;
    massout400(i)=mass400;
    massout600(i)=mass600;
    massout800(i)=mass800;
end

t=0:dt:22.06;
t=t';
massout=massout*1000; %kg to grams
massout400=massout400*1000;
massout600=massout600*1000;
massout800=massout800*1000;
massoutroom=massoutroom*1000;
massoutroomfull=massoutroom*2; %full scale
scount=1;
s6=1;
s4=1;
s2=1;
for m=621:1:1001
    massoutroomsteady(scount)=massoutroomfull(m);
scount=scount+1;
end

for n=671:1:1001
    massout600steady(s6)=massout600(n);
s6=s6+1;
end

for p=411:1:1001
    massout400steady(s4)=massout400(p);
s4=s4+1;
end

for q=481:1:1001
    massout200steady(s2)=massout(q);
s2=s2+1;
end

massout800steady=massout800(1521:2207);

tsteadyroom=10:dt:22;
tsteady200=4.8:dt:10;
tsteady400=4.1:dt:10;
tsteady600=6.7:dt:10;
tsteady800=t(1521:2207);
trend200=fit(tsteady200',massout200steady','poly1');
trend400=fit(tsteady400',massout400steady','poly1');
trend600=fit(tsteady600',massout600steady','poly1');
trend800=fit(tsteady800,massout800steady','poly1');
trendroom=fit(tsteadyroom',massoutroom(1001:2201)','poly1');
trendroomfull=fit(tsteadyroom',massoutroomfull(1001:2201)','poly1');

%%
a = figure('Position', [1 1 3.25 3.25]);
axes('Position', [0.5 0.5 2.5 2.5]);
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',11);

%plot(Data002(:,1),Data002(:,2),'-k','LineWidth',1);
hold on
plot(t,massoutroomfull,:k','LineWidth',1)
hold on
%axis([0 10 0 100])
fmt_pubfig('t\{,\} \: \rm{s}','m \{,\} \: \rm{g}');
%legend('Experiment','Model','Location','Northwest')
%legend boxoff

b = figure('Position', [1 1 3.5 3.5]);
axes('Position', [0.5 0.5 2.5 2.5]);
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',11);
A.5.2 Flow Behaviour Contours

%% Load and Setup parameters
clear all
close all
clc

% read in the data
dataroomsing=csvread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\regionroom\roomsteady_10s.csv',1,1);
dataroom=csvread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\regionroom\roomsteadyave_2.csv',1,1);
mesh=stlread('C:\Users\yarrj\OneDrive - Georgia Institute of Technology\Generation 3\Y2Q3\LIGGGHTS\Paper 2 Simulations\incline_nowalls.stl');
xroom=dataroom(:,1);
yroom=dataroom(:,2);
zroom=dataroom(:,3);
pointsroom=[xroom, yroom, zroom];
velroom=[dataroom(:,13),dataroom(:,14),dataroom(:,15)];
radiusroom=dataroom(:,11); % in meters
%radiusroom=radiusroom*39.3701; % in inches
[lroom wroom]=size(velroom);

pointsroomsing=[dataroomsing(:,1), dataroomsing(:,2),
dataroomsing(:,3)];
velroomsing=[dataroomsing(:,13),dataroomsing(:,14),dataroomsing(:,15)];
radiusroomsing=dataroomsing(:,11);
[lroomsing wroomsing]=size(velroomsing);
for i=1:lroom
vmagroom(i) = (velroom(i,1)^2 + velroom(i,2)^2 + velroom(i,3)^2)^(1/2);
end

vmagroom = vmagroom';

for i = 1:lroomsing

vmagroomsing(i) = (velroomsing(i,1)^2 + velroomsing(i,2)^2 + velroomsing(i,3)^2)^(1/2);
end

vmagroomsing = vmagroomsing';

set(0, 'defaulttextinterpreter', 'latex');
set(0, 'DefaultAxesUnits', 'inches');
set(0, 'DefaultFigureUnits', 'inches');
set(gca, 'Fontname', 'Times New Roman');
set(gca, 'Fontsize', 11); % Number density contours
lcombine = 25; % change this to the number of timesteps averaged
dp = 410*10^(-6); % mean particle diameters
steplength = 9*dp; % Controls resolution
stepwidth = steplength;
gridlength = ceil((0.311391)/steplength);
gridwidth = ceil((0.03978)/stepwidth);
gridx = 0:steplength:0.311391;
gridz = 0:stepwidth:0.03978;

b = figure('Position', [1 1 3.5 3.5]);
axes('Position', [0.25 0.25 3 3]);
h = binscatter(pointsroom(:,1), pointsroom(:,3), [gridlength-1 gridwidth-1], 'ShowEmptyBins', 'On', 'xlim', [0 0.311391]);
xedges = h.XBinEdges;
zedges = h.YBinEdges;
umdensecounts = h.Values;

maxcount = max(numdensecounts(:));
partnum = zeros(gridlength-1, gridwidth-1, maxcount);
for j = 1:1:gridlength-1
    for k = 1:1:gridwidth-1
        if j == gridlength
            if k == gridwidth
                row = find(pointsroom(:,1) >= xedges(j) & pointsroom(:,3) >= zedges(k));
            else
                row = find(pointsroom(:,1) >= xedges(j) & pointsroom(:,3) <= zedges(k) & pointsroom(:,3) < zedges(k+1));
        else
            row = find(pointsroom(:,1) >= xedges(j) & pointsroom(:,3) <= zedges(k));
        end
    end
end
if k==gridwidth
    row=find(pointsroom(:,1)>=xedges(j) & pointsroom(:,1) < xedges(j+1) & pointsroom(:,3) >= zedges(k));
else
    row=find(pointsroom(:,1)>=xedges(j) & pointsroom(:,1) < xedges(j+1) & pointsroom(:,3) >= zedges(k) & pointsroom(:,3) < zedges(k+1));
end

if isempty(row)==1
    partnum(j,k,:)=0;
else
    if length(row) ~= length(unique(row))
        return
    end
    len=size(row);
    for n=1:1:len
        partnum(j,k,n)=row(n);
    end
end

particlevol=zeros(gridlength-1,gridwidth-1);
heightmax=zeros(gridlength-1,gridwidth-1);
heightmaxave=zeros(gridlength-1,gridwidth-1,lcombine);
vmagave=zeros(gridlength-1,gridwidth-1);
avevol=zeros(gridlength-1,gridwidth-1);
heightave=zeros(gridlength-1,gridwidth-1);
for q=1:1:gridlength-1
    for r=1:1:gridwidth-1
        for s=1:1:maxcount
            if partnum(q,r,s)==0
            else
                particlevol(q,r)=particlevol(q,r)+(4/3*pi*radiusroom(partnum(q,r,s),1)^3);
end
            if (pointsroom(partnum(q,r,s),2)+radiusroom(partnum(q,r,s),1)) > heightmax(q,r),
            %if
        end
    end
end
heightmax(q,r)=(pointsroom(partnum(q,r,s),2)+radiusroom(partnum(q,r,s),1)); end

vmagave(q,r)=vmagave(q,r)+vmagroom(partnum(q,r,s),1);

heightave(q,r)=heightave(q,r)+pointsroom(partnum(q,r,s),2);
end

if (nnz(partnum(q,r,:)))==0
avevol(q,r)=0;
vmagave(q,r)=0;
heightave(q,r)=0;
heightmaxave(q,r,:)=0;
else
avevol(q,r)=particlevol(q,r)/(nnz(partnum(q,r,:)));
vmagave(q,r)=vmagave(q,r)/(nnz(partnum(q,r,:)));
heightave(q,r)=heightave(q,r)/(nnz(partnum(q,r,:)));

parttemp=nonzeros(partnum(q,r,:));
ltemp=length(parttemp);
if ltemp < lcombine
heightmaxave(q,r,1:ltemp)=(pointsroom(parttemp(:,2)+radiusroom(parttemp(:,1)));
elseif ltemp == lcombine
heightmaxave(q,r,:)=(pointsroom(parttemp(:,2)+radiusroom(parttemp(:,1)));
else
heightmaxave(q,r,:)=maxk(pointsroom(parttemp(:,2)+radiusroom(parttemp(:,1)),1),lcombine);
end
end

heightmax(q,r)=mean(heightmaxave(q,r,:));
end
end

heightmaxset=(max(heightmax(:))-0.0469679);
unitvolume=steplength*stepwidth*heightmaxset*lcombine; %averaged over lcombine number of separate time steps
numdense=(numdensecounts/(unitvolume));

volfrac=particlevol/unitvolume;
save('contourparametersroom_3.mat','volfrac','heightmax','heightmaxset','avevol','vmagave')
% grid velocity magnitude
yref=0.0469679; %y-coordinate of the particle-wall inclined surface
numlayers=ceil(heightmaxset/(0.0002356*2));
gridheight=numlayers;
yedges=0:0.0002356*2:heightmaxset;
\%split partnum
velmag=zeros(gridlength-1,gridwidth-1,gridheight-1);
velsurface=zeros(gridlength-1,gridwidth-1);
for q=1:1:gridlength-1
   for r=1:1:gridwidth-1
      for tt=1:1:gridheight-1
         col=find(pointsroom(:,1)>=xedges(q) & pointsroom(:,1) <
                  xedges(q+1) & pointsroom(:,3) >= zedges(r) & pointsroom(:,3) <
                  zedges(r+1) & (pointsroom(:,2)-yref) >= yedges(tt) &
                  (pointsroom(:,2)-yref) < yedges(tt+1));
         if isempty(col)=1
            velmag(q,r,tt)=0;
         else
            if length(col) ~= length(unique(col))
               return
            end
            lenc=size(col);
            velmag(q,r,tt)=sum(vmagroom(col(:),1))/lenc(1);
         end
      end
      for tt=gridheight-1:-1:1
         if velmag(q,r,tt) ~= 0
            if tt==1
               velsurface(q,r)=velmag(q,r,tt);
            break
            else
               velsurface(q,r)=(velmag(q,r,tt)+velmag(q,r,tt-1))/2;
            break
            end
         end
      end
   end
end
save('contourparametersroom_4.mat','velmag','velsurface')
\% velsurface plot
m = figure('Position',[1 1 6.5 3.5]);
axes('Position',[0.25 0.25 6 3]);
[Cv,Hv]=contour(velsurface,[0 0.04 0.06 0.07 0.09 0.12 0.16 0.2]);
axis equal
colormap(copper)
caxis([0 0.1]);
lv=clabel(Cv,'manual','Fontsize',11,'FontName','Times New Roman');
%imagesc(velsurface)
ax=gca;
ax.BoxStyle='full';
set(gca,'YDir','normal')

colorbar('off');

%xlabel(""")
ylabel(""")
set(gca,'XTickLabel',{"","","","","",""},
set(gca,'YTickLabel',{"","","","","",""},
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
set(gca,'xtick',[],'ytick',[],'ztick',[]);
hold on

camorbit(90,0,'data',[0 0 1])
% axes('Position',[0.375 0.5 5.75 2],'Color','none','YColor','none');
% set(gca,'XTickLabel',{"","","","","",""},
% set(gca,'XColor', 'none');
% e=colorbar('southoutside');
% caxis([0 0.28]);
% set(e,'FontSize',11,'FontName','Times New Roman','Ticks',[0 0.04 0.08 0.12 0.16 0.2 0.24 0.28])
% e.Label.Interpreter = 'latex';
% e.Label.String="$$\it{V_{\rm{m}}} \: \rm{m \cdot s^{-1}}$$";
print(m,'-dtiff','-r600','velsurface600_steadyave')

%% volfrac and vmagave plots
% c = figure('Position',[1 1 6.5 3.5]);
axes('Position',[0.25 0.25 6 3]);
[Cvf,Hvf]=contour(volfrac,[0 0.07 0.18 0.21 0.28 0.35]);
axis equal
colormap(copper);
caxis([0 0.35]);
 lvf=clabel(Cvf,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(volfac)
set(gca,'YDir','normal')
camorbit(90,0,'data',[0 0 1])
colorbar('off');

%xlabel(""")
ylabel(""")
set(gca,'XTickLabel',{"","","","","",""},
set(gca,'YTickLabel',{"","","","","",""},
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
set(gca,'xtick',[],'ytick',[],'ztick',[]);
hold on
g = figure('Position',[1 1 6.5 3.5]);
axes('Position',[0.25 0.25 6 3]);
[Cvm,Hvm]=contour(vmagave,[0 0.015 0.03 0.04 0.06 0.08 0.09 0.11 0.12 0.14]);
caxis([0 0.15]);
axis equal
colorbar('off');
hold on
set(gca,'XTickLabel',""');
set(gca,'YTickLabel',""');
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
set(gca,'xtick',[],'ytick',[],'ztick',[]);
hold off

% axes('Position',[0.375 0.5 5.75 2],'Color','none','YColor','none');
% set(gca,'XTickLabel',""');
% set(gca,'YColor', 'none');
% e=colorbar('southoutside');
% caxis([0 0.35]);
% set(e,'FontSize',11,'FontName','Times New Roman','Ticks',[0 0.07 0.14 0.21 0.28 0.35])
% e.Label.Interpreter = 'latex';
% e.Label.String="$\it{\varphi \rm{,} \: \rm{-}}$";
% %print(c,'-dtiff','-r600','volfrac800_steadyave')

d = figure('Position',[1 1 6.5 3.5]);
axes('Position',[0.25 0.25 6 3]);
[Cbh,Hbh]=contour(bedthickness,[0 0.50 0.6 0.7 0.8 0.9 1]);
caxis([0 1]);
colormap(copper);
axis equal
lbh=clabel(Cbh,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(bedthickness)
set(gca,'YDir','normal')

colorbar('off');

%xlabel("")
ylabel("")
set(gca,'XTickLabel',{"","","","","","",""});
set(gca,'YTickLabel',{"","","","","","",""});
set(gca,'XColor','Black','YColor','Black','ZColor','Black');
set(gca,'xtick',[],'ytick',[],'ztick',[]);
hold on
camorbit(90,0,'data',[0 0 1])

% axes('Position',[0.375 0.5 5.75 2],'Color','none','YColor','none');
% set(gca,'XTickLabel',{"","","","","","",""});
% set(gca,'XColor','none');
% e=colorbar('southoutside');
% set(e,'FontSize',11,'FontName','Times New Roman')
% e.Label.Interpreter = 'latex';
% e.Label.String="\it{\widetilde{h} \text{,} \text{-}}";
caxis([0 1]);
print(d,'-dtiff','-r600','bedheight800_steadyave')

%% mass flux
massflux=volfrac*3270.*vmagave;

f = figure('Position',[1 1 6.5 3.5]);
axes('Position',[0.25 0.25 6 3]);
[CM,HM]=contour(massflux,[0 15 30 40 60 70]);
colormap(copper);
axis equal
caxis([0 80]);
lm=clabel(CM,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(massflux)
set(gca,'YDir','normal')

colorbar('off');

%xlabel("")
ylabel("")
set(gca,'XTickLabel',{"","","","","","",""});
set(gca,'YTickLabel',{"","","","","","",""});
set(gca,'XColor','Black','YColor','Black','ZColor','Black');
set(gca,'xtick',[],'ytick',[],'ztick',[]);
hold on
camorbit(90,0,'data',[0 0 1])

% axes('Position',[0.375 0.5 5.75 2],'Color','none','YColor','none');
% set(gca,'XTickLabel',{'';''';''';''';''';''';});
% set(gca,'XColor', 'none');
% e=colorbar('southoutside');
% caxis([0 100]);
% set(e,'FontSize',11,'FontName','Times New Roman')%,'Ticks',[0 12.5,25.0,37.5,50.0,62.5,75.0])
% e.Label.Interpreter = 'latex';
% e.Label.String="\it{\dot{m}'' \: \: \rm{kg \cdot m^{-2} \cdot s^{-1}}}$$";
print(f,'-dtiff','-r600','massflux800_steadyave')

A.5.2.1 Combined Contour Figures

clear all
close all
clc

set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(0,'DefaultFigureUnits','inches');
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',11);

stepwidth=9*dp; %Controls resolution

set(0,'defaulttextinterpreter','latex');
set(0,'DefaultAxesUnits','inches');
set(gca, 'Fontname','Times New Roman');
set(gca, 'Fontsize',11);

% velsurface plots
load('contourparametersroom_3.mat')
load('contourparametersroom_4.mat')

m = figure('Position', [1 1 6.5 6]);
axes('Position', [0.5 3.875 5.5 3]);
imagesc(fliplr(gridz),gridx,velsurface)
caxis([0 0.24]);
colormap(jet)

%set(gca,'YDir','reverse')
hold on
[Cv,Hv]=contour(fliplr(gridz),gridx,velsurface,[0 0.04 0.08 0.12 0.14 0.16 0.2],'LineWidth',0.5,'Color','k');
axis equal
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off','YDir','reverse')
set(gca,'YDir','normal')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
colorbar('off');
fmt_pubfig('W \text{,} \text{mm}','L \text{,} \text{mm}')
ylabel(""")
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(a)')
hold on

load('contourparameters200_3.mat')
load('contourparameters200_4.mat')

axes('Position',[0.5 2.75 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,velsurface,[0 0.04 0.08 0.12 0.16 0.2 0.22],['LineWidth',1.5]);
axis equal
colormap(jet)
caxis([0 0.24]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
box off
colorbar('off');
fmt_pubfig('W \text{,} \text{mm}','L \text{,} \text{mm}')
ylabel(""")
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(b)')
hold on

load('contourparameters400_3.mat')
load('contourparameters400_4.mat')

axes('Position',[0.5 1.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,velsurface,[0 0.04 0.08 0.12 0.16 0.2 0.22],['LineWidth',1.5]);
axis equal
colormap(copper)
caxis([0 0.24]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \text{,} \text{mm}','L \text{,} \text{mm}')
ylabel(""")
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(c)')
hold on
load('contourparameters600_3.mat')
load('contourparameters600_4.mat')

axes('Position',[0.5 0.5 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,velsurface,[0 0.04 0.08 0.12 0.16 0.2],'
'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.24]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \rm{,} \: \rm{mm}','L \rm{,} \: \rm{mm}')
ylabel('''
set(gca,'XColor', 'Black', 'YColor', 'Black', 'ZColor', 'Black');
camorbit(90,0,'data',[0 0 1])
title('(d)')
hold on

load('contourparameters800_3.mat')
load('contourparameters800_4.mat')

axes('Position',[0.5 -0.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,velsurface,[0 0.04 0.06 0.07 0.09 0.12 0.16 0.2],'
'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.24]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \rm{,} \: \rm{mm}','L \rm{,} \: \rm{mm}')
ylabel('''
set(gca,'XColor', 'Black', 'YColor', 'Black', 'ZColor', 'Black');
camorbit(90,0,'data',[0 0 1])
title('(e)')
hold on

load('contourparameters800_3.mat')
load('contourparameters800_4.mat')

axes('Position',[0.5 -0.625 5.5 3]);

[c = figure('Position',[1 1 6.5 6]);
load('contourparametersroom_3.mat')
load('contourparametersroom_4.mat')

Cv,Hv]=contour(fliplr(gridz),gridx,volfrac,[0 0.07 0.14 0.21 0.28 0.35],'
'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.35]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')

%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \text{,} \: \text{mm}','L \text{,} \: \text{mm}')
ylabel('')

load('contourparameters200_3.mat')
load('contourparameters200_4.mat')

axes('Position',[0.5 2.75 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,volfrac,[0 0.07 0.14 0.18 0.21 0.28 0.35],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.35]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
 lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)

box off

colorbar('off');
fmt_pubfig('W \text{,} \: \text{mm}','L \text{,} \: \text{mm}')
ylabel('')
camorbit(90,0,'data',[0 0 1])
title('(a)')
hold on

load('contourparameters400_3.mat')
load('contourparameters400_4.mat')

axes('Position',[0.5 1.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,volfrac,[0 0.07 0.14 0.18 0.21 0.28 0.35],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.35]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
 lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)

colorbar('off');
fmt_pubfig('W \text{,} \: \text{mm}','L \text{,} \: \text{mm}')
ylabel('')
camorbit(90,0,'data',[0 0 1])
title('(b)')
hold on

load('contourparameters600_3.mat')

104
load('contourparameters600_4.mat')

axes('Position',[0.5 0.5 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,volfrac,[0 0.07 0.14 0.18 0.21 0.28 0.35],''LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.35]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,manual,'FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \, \text{mm}','L \, \text{mm}')
ylabel('')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(d)')
hold on

load('contourparameters800_3.mat')
load('contourparameters800_4.mat')

axes('Position',[0.5 -0.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,volfrac,[0 0.07 0.18 0.21 0.28 0.35],''LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.35]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,manual,'FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \, \text{mm}','L \, \text{mm}')
ylabel('')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(e)')
hold on

load('contourparameters800_room_4.mat')

massflux=volfrac*3270.*vmagave;
f = figure('Position',[1 1 6.5 6]);
axes('Position',[0.5 3.875 5.5 3]);
camorbit(90,0,'data',[0 0 1])

print(c,'-dtiff','-r600','volfracfinal')

%% mass flux
set(0,'defaulttextinterpreter','latex');
load('contourparametersroom_3.mat')
load('contourparametersroom_4.mat')
massflux=volfrac*3270.*vmagave;
f = figure('Position',[1 1 6.5 6]);
axes('Position',[0.5 3.875 5.5 3]);
camorbit(90,0,'data',[0 0 1])
[Cv,Hv]=contour(fliplr(gridz),gridx,massflux,[0 15 30 45 55 60 65],''LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 80]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
load('contourparameters200_3.mat')
load('contourparameters200_4.mat')
massflux=volfrac*3270.*vmagave;
axes('Position',[0.5 2.75 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,massflux,[0 15 30 45 60 70],'
LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 80]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
box off
colorbar('off');
fmt_pubfig(W \rm{,} \: \rm{mm},'L \rm{,} \: \rm{mm}')
ylabel('
set(gca,'XColor', 'Black', 'YColor', 'Black', 'ZColor', 'Black');
camorbit(90,0,'data',[0 0 1])
title('(a)')
hold on

load('contourparameters400_3.mat')
load('contourparameters400_4.mat')
massflux=volfrac*3270.*vmagave;
axes('Position',[0.5 1.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,massflux,[0 15 30 45 60 70],'
LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 80]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
box off
colorbar('off');
fmt_pubfig(W \rm{,} \: \rm{mm},'L \rm{,} \: \rm{mm}')
ylabel('
set(gca,'XColor', 'Black', 'YColor', 'Black', 'ZColor', 'Black');
camorbit(90,0,'data',[0 0 1])
title('(b)')
hold on

load('contourparameters600_3.mat')
load('contourparameters600_4.mat')
massflux=volfrac*3270.*vmagave;
axes('Position',[0.5 0.5 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,massflux,[0 15 30 45 60 70],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');

%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \text{,} \: \text{mm}','L \text{,} \: \text{mm}')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(d)')
hold on

load('contourparameters800_3.mat')
load('contourparameters800_4.mat')

massflux=volfrac*3270.*vmagave;
axes('Position',[0.5 -0.625 5.5 3]);
[Cv,Hv]=contour(fliplr(gridz),gridx,massflux,[0 15 30 40 60 70],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')

%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \text{,} \: \text{mm}','L \text{,} \: \text{mm}')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(e)')

print(f,'-dtiff','-r600','massfluxfinal')

%% Bed thickness
set(0,'defaulttextinterpreter','latex');
load('contourparametersroom_3.mat')
load('contourparametersroom_4.mat')
yref=0.0469679;
bedthickness=(heightmax-yref)./(heightmaxset);
d = figure('Position',[1 1 6.5 6]);
axes('Position',[0.5 3.875 5.5 3]);
camorbit(90,0,'data',[0 0 1])
[Cv,Hv]=contour(fliplr(gridz),gridx,bedthickness,[0 0.40 0.5 0.6 0.65 0.8 1],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 1]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')

%imagesc(velsurface)
\begin{verbatim}
[Cv,Hv]=contour(fliplr(gridz)),gridx,bedthickness,[0 0.45 0.5 0.55 0.6 
0.8 1],['LineWidth',1.5); axis equal colormap(copper) caxis([0 1]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
imagesc(velsurface) colorbar('off');
fmt_pubfig('W \, \: \, \, L \, \: \, \, \, \, \, mm') ylabeled('"')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1]) title('(d)')
hold on
load('contourparameters800_3.mat')
load('contourparameters800_4.mat')
bedthickness=(heightmax-yref)./(heightmaxset);
axes('Position',[0.5 -0.625 5.5 3]);
[Cv,Hv]=contour(fliplr(gridz),gridx,bedthickness,[0 0.50 0.6 0.7 0.8 
1],['LineWidth',1.5); axis equal colormap(copper) caxis([0 1]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
imagesc(velsurface) colorbar('off');
fmt_pubfig('W \, \: \, \, L \, \: \, \, \, \, \, \, mm') ylabeled('"')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1]) title('(e)')
print(d,'-dtiff','-r600','bedheightfinal')
\end{verbatim}
camorbit(90,0,'data',[0 0 1])
title('(a)')
hold on
load('contourparameters200_3.mat')
load('contourparameters200_4.mat')

axes('Position',[0.5 2.75 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,vmagave,[0 0.03 0.09 0.11 0.12 0.14],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.15]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')

lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W rm(,)
L rm(,) \L rm(mm)')
ylabel('')
set(gca,'XColor','Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(b)')
hold on

load('contourparameters400_3.mat')
load('contourparameters400_4.mat')

axes('Position',[0.5 1.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,vmagave,[0 0.03 0.09 0.11 0.12 0.14],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.15]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')

lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W rm(,)
L rm(,) \L rm(mm)')
ylabel('')
set(gca,'XColor','Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(c)')
hold on

load('contourparameters600_3.mat')
load('contourparameters600_4.mat')

axes('Position',[0.5 0.5 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,vmagave,[0 0.03 0.06 0.08 0.10],'LineWidth',1.5);
axis equal
load('contourparameters800_3.mat')
load('contourparameters800_4.mat')

axes('Position',[0.5 -0.625 5.5 3]);

[Cv,Hv]=contour(fliplr(gridz),gridx,vmagave,[0 0.015 0.03 0.04 0.06],'LineWidth',1.5);
axis equal
colormap(copper)
caxis([0 0.15]);
set(gca,'YAxisLocation','Right','XDir','reverse','XMinorTick','Off')
lv=clabel(Cv,'manual','FontSize',11,'FontName','Times New Roman');
%imagesc(velsurface)
colorbar('off');
fmt_pubfig('W \text{,} \text{mm}','L \text{,} \text{mm}')
ylabel('')
set(gca,'XColor', 'Black','YColor','Black','ZColor','Black');
camorbit(90,0,'data',[0 0 1])
title('(d)')
hold on

print(d,'-dtiff','-r600','avevelocityfinal')
REFERENCES


