INTERNATIONAL UNIVERSITY LIAISON INDONESIA (IULI)

BACHELOR'S THESIS

SIMULATION OF EJECTA DYNAMICS ON RYUGUS SURFACE INDUCED BY VERTICAL IMPACTS USING YADE

By

Talal Mohammed Ali Bin Snkar

11202101012

Presented to the Faculty of Engineering and Life Sciences

In Partial Fulfilment Of the Requirements for the Degree of

SARJANA TEKNIK

In

AVIATION ENGINEERING

FACULTY OF ENGINEERING AND LIFE SCIENCES

BSD City 15345

Indonesia

January 2025

APPROVAL PAGE

SIMULATION OF EJECTA DYNAMICS ON RYUGUS SURFACE INDUCED BY VERTICAL IMPACTS USING YADE

TALAL MOHAMMED ALI BIN SNKAR
11202101012
Presented to the Faculty of Engineering
In Partial Fulfillment of the Requirements for the Degree of

SARJANA TEKNIK
In
AVIATION ENGINEERING

FACULTY OF ENGINEERING AND LIFE SCIENCES

Triwanto Simanjuntak, PhD	____________________
Thesis Advisor	Date (MM/DD/YYYY)
Dr. Eng. Ressa Octavianty	____________________
Thesis Co-Advisor	Date (MM/DD/YYYY)
Dipl.-Ing. Sentot Wahjoe Georitno, M.Si.	____________________
Dean of Faculty of Engineering and Life Sciences	Date (MM/DD/YYYY)

EXAMINERS APPROVAL PAGE

Dr. Eng. Odi Akhyarsi
Examiner 1

Date

M. Ridlo Erdata Nasution, PhD
Examiner 2

Date

STATEMENT BY THE AUTHOR

I hereby declare that this submission is my own work and to the best of my knowledge, it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at any educational institution, except where due acknowledgement is made in the thesis.

Talal Mohammed Ali Bin Snkar
Student

Date (MM/DD/YYYY)

ABSTRACT

Simulation of Ejecta Dynamics on Ryugus Surface Induced by Vertical Impacts Using YADE

by

Talal Mohammed Ali Bin Snkar

Triwanto Simanjuntak, PhD, Advisor

Dr. Eng. Ressa Octavianty, Co-Advisor

Understanding ejecta dynamics on asteroid surfaces is crucial for studying planetary formation and designing space exploration missions. This research focuses on simulating the motion of regolith particles on the asteroid Ryugu using the discrete element method (DEM) in YADE, addressing challenges in modeling regolith by incorporating polyhedral particle geometries, which enhance realism compared to traditional spherical models. Numerical simulations were conducted under low-gravity conditions (0.00011 $\mathrm{m/s^2}$), with an impact velocity of 300 $\mathrm{m/s}$ and particle sizes ranging from 1 to 10 mm, analyzing key metrics such as velocity decay, particle trajectories, and energy dissipation. Results demonstrate that polyhedral particles exhibit higher energy dissipation and more accurate velocity decay patterns, closely aligning with theoretical models of ejecta behavior under microgravity. This study advances computational modeling of asteroid surface dynamics, providing insights into regolith response during impact events with implications for mission planning, including sampling, anchoring mechanisms, and planetary defense strategies. Future research should explore multi-impact scenarios and incorporate external forces like solar radiation to improve model fidelity.

Keyword: *Asteroid Regolith, Ejecta Dynamics, Discrete Element Method, Microgravity Simulation, Polyhedral Particles*

ACKNOWLEDGEMENTS

Firstly, I express my deepest gratitude to Allah Almighty for His blessings and guidance in completing this thesis. Without His assistance, this accomplishment would not have been possible.

I am profoundly grateful to my advisor, Triwanto Simanjuntak, Ph.D., and my co-advisor, Dr. Eng. Ressa Octavianty, for their invaluable guidance, unwavering support, and continuous encouragement throughout the course of my research. Their mentorship was instrumental in helping me complete this thesis, and I cannot thank them enough for their dedication.

I would also like to extend my sincere appreciation to my colleagues from my batch for their constant help and support. A special thanks to Samudra Prasetyo and Karol Brzezinski for their insightful comments, thoughtful suggestions, and unwavering assistance during my thesis work. Your support has been truly invaluable.

I am especially grateful to my examiners, Dr. Eng. Odi Akhyarsi and M. Ridlo Erdata Nasution, Ph.D., for their time, constructive feedback, and valuable insights, which have significantly contributed to improving this research. Their expertise and thoughtful discussions during my defense have helped refine and strengthen my thesis.

Lastly, I am deeply indebted to my parents and family for their unconditional love, prayers, and support, which have been my constant source of strength. I also thank my friends and brothers who stood by me during my studies and thesis journey. Your encouragement means the world to me.

Thank you all.

Contents

List of Figures

List of Tables

List of Abbreviations

DEM	**D**iscrete **E**lement **M**ethod
YADE	**Y**et **A**nother **D**ynamic **E**ngine
NEOs	**N**ear **E**arth **O**bjects
SCI	**S**mall **C**arry-on **I**mpactor
TAGSAM	**T**ouch **A**nd **G**o **S**ample **A**cquisition **M**echanism
GUI	**G**raphical **U**ser **I**nterface

Dedicated to my parents

CHAPTER 1
INTRODUCTION

1.1 Background

Asteroids are remnants from the early formation of the solar system, preserving clues about the building blocks of planets and the environment of the early solar system. Deep space missions targeting asteroids aim to unlock these secrets by studying their surface compositions and dynamics. Missions such as Hayabusa, OSIRIS-REx, and Hayabusa2 have significantly contributed to planetary science by returning samples and providing insights into asteroid structure and regolith behavior. In addition to their scientific importance, these missions play a crucial role in planetary defense by improving models of impact dynamics and strategies for mitigating potential asteroid collisions with Earth see Figure **??**.

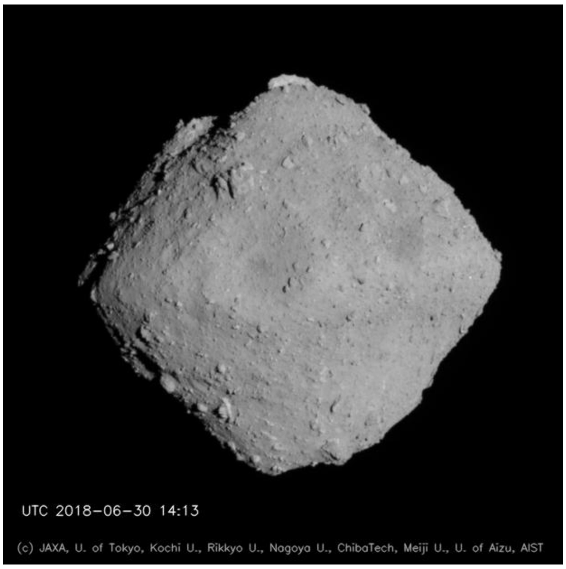


FIGURE 1.1: Ryugu Asteroid **AsteroidExplorerHayabusa2**.

Understanding asteroid ejecta behavior is fundamental for surface evolution models and mission design. This research uses the Discrete Element Method (DEM) implemented in YADE (Yet Another Dynamic Engine) to simulate and analyze particle interactions under low-gravity conditions representative of small asteroids like Ryugu and Itokawa. DEM simulations allow the study of regolith particle motion, impact-induced ejecta trajectories, and energy dissipation, which are vital for advancing planetary exploration and defense applications.

1.2 Problem Statement

Asteroid surface processes are governed by interactions between regolith particles under microgravity conditions. However, modeling ejecta behavior in such environments remains a challenge due to the complexity of particle interactions, shape irregularities, and computational constraints. Existing models often oversimplify particle shapes, ignoring the significant effects of angular and polyhedral geometries on regolith behavior. Additionally, the influence of weak gravity on particle trajectories and energy dissipation needs further investigation.

This research addresses these gaps by simulating ejecta motion on an asteroid-like surface using DEM with YADE, incorporating polyhedral particles to enhance realism. The results will provide a better understanding of surface dynamics and contribute to the design of future missions and planetary defense strategies see Figure **??**.

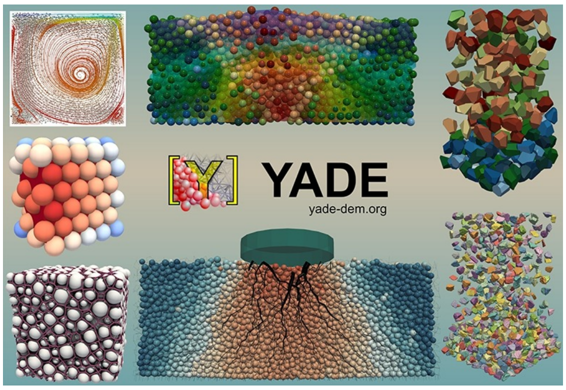


FIGURE 1.2: YADE (Yet Another Dynamic Engine) **angelidakisYADEExtensibleFramework2024**.

1.2.1 Deep Space Exploration Missions

Asteroids are remnants from the early formation of the solar system, making them natural archives that preserve clues about the primordial conditions of planetary formation. Studying these ancient bodies provides critical insights into the origins of our solar system and the building blocks of terrestrial planets. Deep space exploration missions targeting asteroids have become essential for understanding their composition, structure, and evolution. In addition to their scientific importance, these missions support planetary defense initiatives by enhancing our knowledge of near-Earth objects (NEOs) and strategies to mitigate potential impact threats see Figures **??**, **??**.

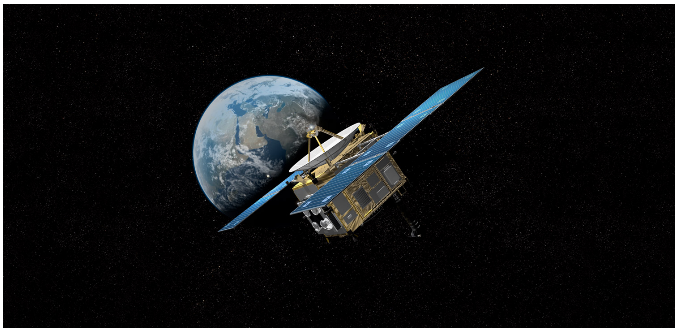


FIGURE 1.3: The Hayabusa spacecraft **HAYABUSASpacecraft**.

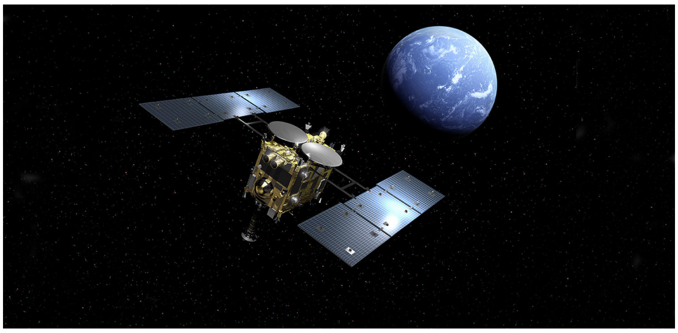


FIGURE 1.4: The Hayabusa2 spacecraft **AsteroidExplorerHayabusa2**.

The technological advancements in spacecraft design and sampling mechanisms have enabled unprecedented exploration of asteroids, providing vital data to address fundamental questions about planetary formation, organic compound distribution, and the role of asteroids in delivering water and life-essential materials to Earth. The significance of asteroid missions lies not only in their contribution to scientific discovery but also in their demonstration of complex navigational and operational technologies for future interplanetary exploration.

1.2.2 Mission Overview

Several landmark missions have targeted asteroids for detailed study and sample collection, significantly advancing our understanding of these primitive celestial bodies. Table **??** summarizes some of the most notable asteroid exploration missions.

Mission	Space Agency	Year	Asteroid	Spacecraft
Hayabusa	JAXA	2003	Itokawa (Fig. **??**)	Hayabusa (Fig. **??**)
OSIRIS-REx	NASA	2016	Bennu (Fig. **??**)	OSIRIS-REx (Fig. **??**)
Hayabusa2	JAXA	2014	Ryugu (Fig. **??**)	Hayabusa2 (Fig. **??**)
Psyche	NASA	2016	16 Psyche See Figure **??**	Psyche (Fig. **??**)

TABLE 1.1: Notable Asteroid Exploration Missions **JAXAJapanAerospace**, **NASA**.

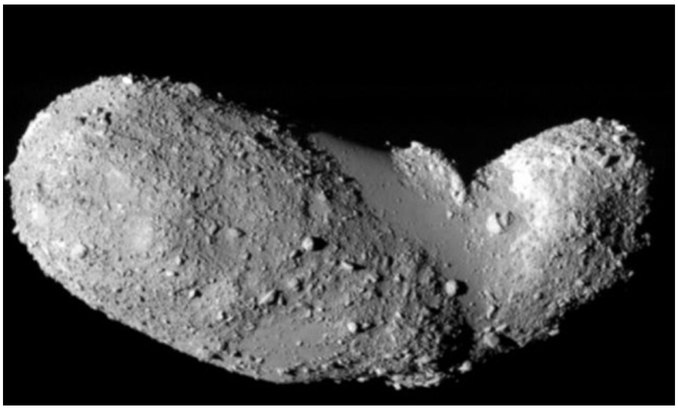


FIGURE 1.5: Itokawa Asteroid **HAYABUSASpacecraft**.

Mission	Mission Overview
Hayabusa	First mission to return asteroid samples to Earth, marking a milestone in sample-return technology.
OSIRIS-REx	Collected a surface sample from Bennu and returned it to Earth in 2023, providing crucial data on carbonaceous asteroids.
Hayabusa2	Successfully returned both surface and subsurface samples from Ryugu in 2020.
Psyche	Aimed at studying a metal-rich asteroid believed to be the exposed core of a protoplanet, providing insights into planetary core formation.

TABLE 1.2: Asteroid Exploration Missions Overviews **JAXAJapanAerospace**, **NASA**.

1.2.3 Sample Collection Techniques

Asteroid missions utilize innovative techniques to overcome the challenges of sample collection in low-gravity environments.

1. Hayabusa and Hayabusa2 (JAXA): The sampling mechanism for both missions used a sampler horn to fire a small projectile into the asteroids surface. The resulting ejecta was captured inside a collection chamber. Additionally, Hayabusa2 deployed the Small Carry-on Impactor (SCI), a device that created an artificial crater to enable subsurface sampling. This approach allowed for the retrieval of regolith unaffected by surface weathering see figures **??**, **??** **HAYABUSASpacecraft**, **AsteroidExplorerHayabusa2**.
2. OSIRIS-REx (NASA): This mission employed the Touch-and-Go Sample Acquisition Mechanism (TAGSAM), designed to briefly contact the asteroids

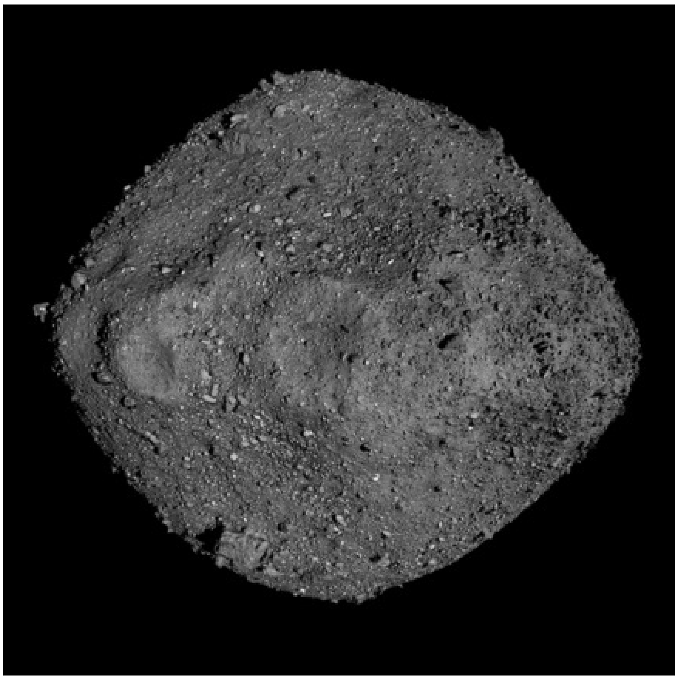


FIGURE 1.6: Bennu Asteroid **OSIRISRExNASAScience2023**.

FIGURE 1.7: 16 Psyche Asteroid **PsycheNASAScience**.

FIGURE 1.8: The Psyche spacecraft near asteroid 16 Psyche **PsycheNASAScience**.

FIGURE 1.9: The OSIRIS-REx spacecraft near asteroid Bennu **OSIRISRExNASAScience2023**.

surface. Nitrogen gas was used to mobilize surface material, which was then captured in a collector head. This innovative design minimized contact time, reducing contamination risks and mechanical failure see figure **?? OSIRISRExNASAScience2023**.

3. Psyche (NASA): Unlike the other missions, Psyche focuses on remote sensing rather than sample return. The spacecraft will use spectrometers, magnetometers, and imagers to study the metal-rich asteroid 16 Psyche, believed to be the core of an early planet **PsycheNASAScience**.

1.3 Research Objectives

The primary objectives of this research are:

1. To simulate ejecta motion on a low-gravity asteroid surface using a discrete element method with polyhedral particle modeling.

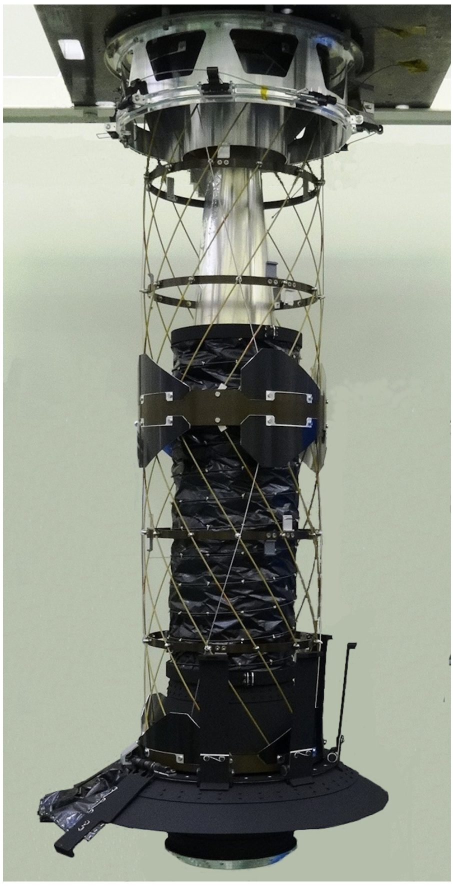


FIGURE 1.10: Shows the sampler horn **AsteroidExplorerHayabusa2**.

2. To analyze the effects of gravity, particle shape, and impact velocity on ejecta trajectories and dispersal patterns.
3. To evaluate the implications of ejecta dynamics for asteroid surface evolution and mission design.
4. To identify computational and physical limitations in simulating complex granular systems with non-spherical particles.

FIGURE 1.11: Samples collected from Ryugu asteroid **AsteroidExplorerHayabusa2**.

FIGURE 1.12: Shows the horn used on OSIRIS-REx spacecraft **OSIRISRExNASAScience2023**.

1.4 Research Scope And Limitation

This study focuses on simulating ejecta behavior on small-body asteroids, specifically under conditions similar to Ryugu or Itokawa. The simulations are performed using YADE, with key parameters including:

- Gravity: A low-gravity environment of approximately $0.00011\ \mathrm{m/s^2}$.
- Particle Shapes: Polyhedral particles ranging from $1\ \mathrm{mm}$ to $10\ \mathrm{mm}$ in size.
- Impact Conditions: An impacting sphere with a velocity of $300\ \mathrm{m/s}$.

The limitations of this research include:

1. Computational Resources: Simulating complex particle interactions with detailed physics requires significant processing power, limiting the scale and duration of simulations.
2. Single Impact Scenario: This study models a single impact event, whereas real asteroid surfaces experience repeated bombardments over time.
3. Simplified External Forces: The effects of solar radiation pressure, cohesion, and electrostatic forces are not included in this analysis.

These constraints highlight areas for future research, including multi-impact simulations, enhanced computational techniques, and the incorporation of additional forces to improve model fidelity.

CHAPTER 2
LITERATURE REVIEW

2.1 Asteroid Characteristics and Surface Composition

2.1.1 Asteroid Structure and Composition

Asteroids have surfaces that are mechanically weathered due to various processes, including impact bombardment and thermal fatigue. These processes alter the composition and size distribution of the surface regolith (the loose material covering the asteroid). Regolith grain sizes are closely tied to an asteroid's thermal inertia and evolution. Larger asteroids with more evolved surfaces tend to have finer grain regolith, whereas smaller or slow-rotating bodies show coarser grains due to less frequent impact weathering **maclennanThermophysicalInvestigationAsteroid**

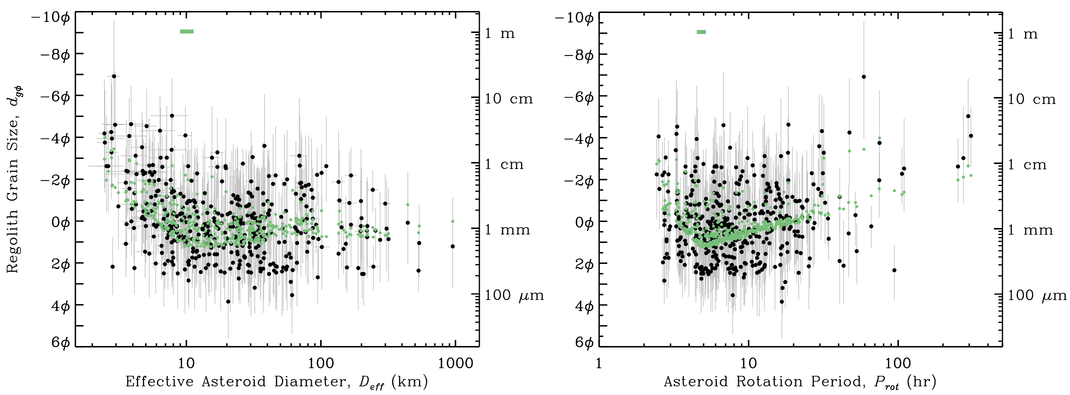


FIGURE 2.1: Highlights the correlation between asteroid diameter and grain size, useful for understanding regolith behavior based on thermal inertia **maclennanThermophysicalInvestigationAsteroid2022**.

As shown in Figure **??**, the relationship between asteroid diameter and regolith grain size is clearly depicted. This figure illustrates how larger asteroids, typically characterized by slower rotation rates and longer exposure to thermal fatigue, develop finer regolith due to prolonged weathering processes. Conversely, smaller asteroids tend to exhibit coarser grain sizes due to their limited thermal evolution and more frequent impact processes.

The thermal conductivity of regolith affects the heat transfer, which in turn influences grain fragmentation through expansion and contraction. Fouriers law for heat conduction describes this thermal interaction:

$$q = -k\frac{dT}{dx} \tag{2.1}$$

Where q is the heat flux, k is the thermal conductivity, and $\frac{dT}{dx}$ is the temperature gradient across the regolith layer **matuttisUnderstandingDiscreteElement2014**. This equation is essential for understanding how thermal properties affect surface fragmentation, impacting regolith particle sizes on asteroid surfaces.

2.1.2 Types of Asteroids and Regolith Properties

Asteroids are broadly categorized into C-type, S-type, and M-type based on their composition, which also affects the nature of the surface regolith and the behavior of ejecta after an impact.

- C-Type Asteroids: Composed mainly of carbonaceous materials, C-type asteroids like Bennu and Ryugu have a dark, porous surface. Their regolith is typically fine-grained and loosely packed, which leads to slower ejecta velocities and wider dispersal patterns during an impact. These asteroids reflect only about 3 percent of incident sunlight, indicative of their low albedo **bartlettDifferentTypesAsteroids2020**. As shown in figure **??** the spatial distribution of craters on Bennus surface reveals key insights into the regoliths fine-grained nature. The figure highlights areas where the regolith exhibits impact armoring, a process where fine particles fill and obscure smaller craters over time.

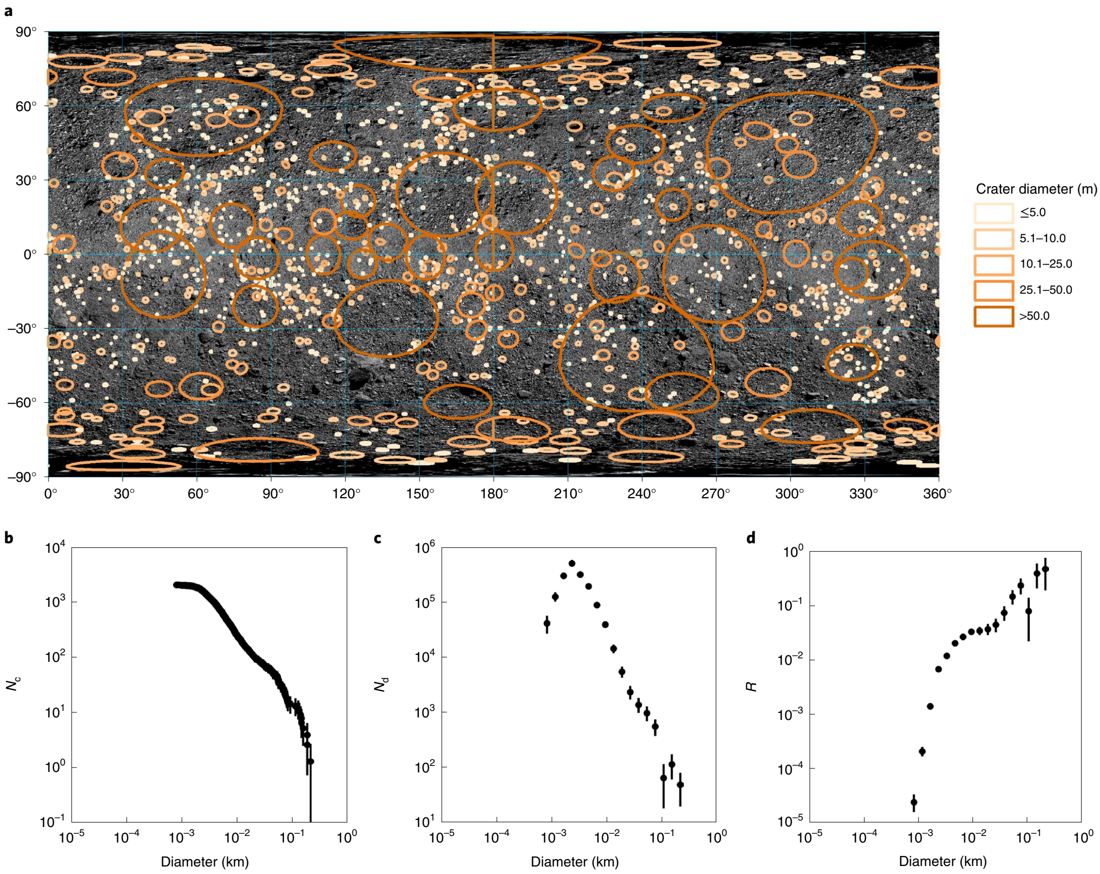


FIGURE 2.2: Regolith and crater distribution on Bennu, showing how fine-grained surfaces affect ejecta distribution **bierhausCraterPopulationAsteroid2022a**.

- S-Type Asteroids: These stony bodies contain silicate materials mixed with nickel-iron. The regolith on S-type asteroids is generally more compact and cohesive, resulting in faster, more localized ejecta dispersal. The NEAR-Shoemaker mission to Eros provides valuable data on how stony asteroid regolith reacts to impact **bartlettDifferentTypesAsteroids2020**.
- M-Type Asteroids: Less common, M-type asteroids are primarily composed of nickel-iron. Their dense surfaces produce energetic, high-velocity ejecta when impacted. The metallic composition can cause more significant heating and vaporization of ejecta, compared to C- or S-type bodies **bartlettDifferentTypesAster**

The behavior of ejecta is influenced by asteroid types and their surface properties. For instance, the power-law relationship between ejecta velocity $v(r)$ and distance from the impact point:

$$v(r) = v_0(\frac{r}{R})^{-\alpha} \tag{2.2}$$

Where:

- v_0 is the initial ejection velocity,
- r is the distance from the impact center,
- R is the crater radius,
- α is an empirically determined constant.

This power law helps model the dispersal patterns of ejecta on C- and S-type asteroids, showing that finer regolith results in slower, wider-spread ejecta **karetaEjectaEvolutionFollowing2023**.

2.2 Impact Cratering Mechanisms

2.2.1 Impact Physics and Cratering Models

Impact cratering on asteroids involves the transfer of kinetic energy from the impacting object (comet or asteroid) to the asteroids surface, resulting in the formation of craters and the ejection of surface material. High-velocity impacts, especially those exceeding $15\ \mathrm{km/s}$, produce complex ejecta patterns. The study **artemievaNumericalSimulationHighvelocity2008a** uses the 3D SOVA hydrocode to model the dynamics of impacts on the Moon, with relevance to asteroid impacts.

As shown in Figure **??** the relationship between impact angle and ejecta mass is vividly depicted. This figure compares the ejecta masses generated by vertical and oblique impacts, highlighting that oblique impacts (e.g., at $45°$) yield significantly more ejecta mass compared to vertical impacts. The visual aids in comprehending how the impact angle influences the material ejected from the crater. The steeper the angle, the more asymmetrical the ejecta distribution becomes, with larger masses being ejected in a specific direction. This observation underscores the importance of considering angle-dependent dynamics when modeling ejecta, particularly for high-energy impacts on asteroid surfaces. By examining this figure, readers can grasp the broader implications of impact geometry on material redistribution in low-gravity environments.

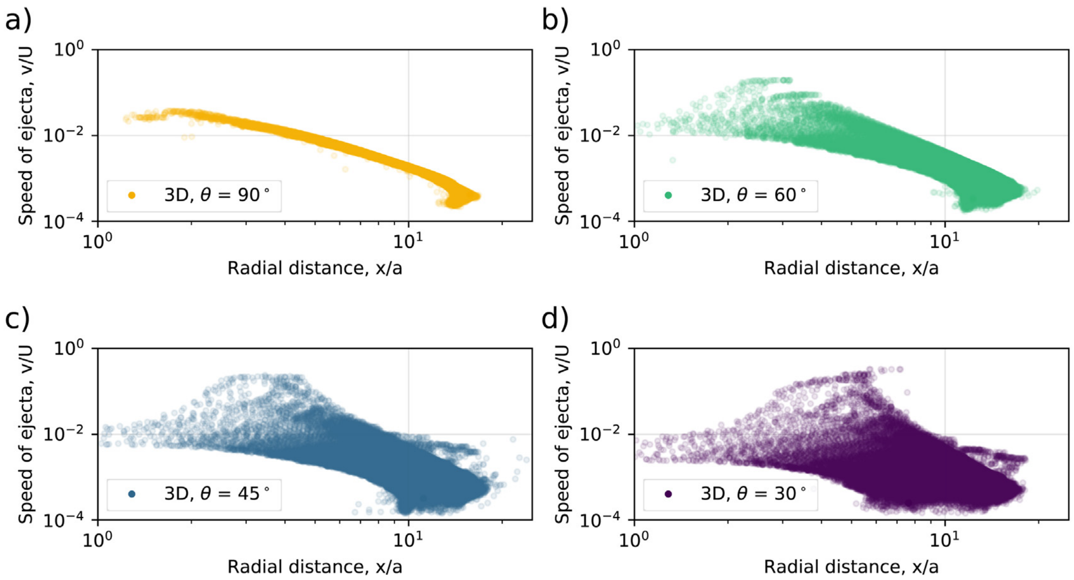


FIGURE 2.3: Illustrates how ejecta mass changes with impact angle, essential for understanding how high-energy impacts vary by angle **RADUCAN2022114793**.

High-velocity impacts produce ejecta with velocities governed by impact dynamics. The ejecta velocity U as a function of distance from the impact point is given by:

$$U = \frac{gR_{cr}}{K_1}(\frac{x}{R_{cr}})^{-e_x} \tag{2.3}$$

Where:

- g is the gravitational acceleration,
- R_{cr} is the crater radius,
- K_1 and e_x are constants dependent on material properties,
- x is the distance from the center.

This equation is valuable for modeling velocity variations across the impact zone, showing how gravity and material properties influence ejecta speed **artemievaNumericalSi**

2.2.2 Vertical vs. Oblique Impacts

The angle of impact significantly influences the morphology of the crater and the distribution of ejecta. Vertical impacts (90°) tend to create symmetric craters and more uniform ejecta blankets, while oblique impacts produce asymmetrical craters and irregular ejecta patterns. The paper **artemievaNumericalSimulationHighvelocity200** demonstrates that oblique impacts (e.g., 45°) yield significantly more ejecta due to the greater surface area affected by the initial shock wave.

As seen in Figure **??** The figure distinctly shows that vertical impacts generate symmetrical craters with a uniform ejecta distribution around the impact site, whereas oblique impacts create asymmetrical craters with ejecta concentrated in a specific direction.

The impact angle impacts both crater shape and ejecta distribution. For vertical impacts, the transient crater radius R_{cr} can be estimated as follows:

$$R_{cr} = K_2(\frac{ga}{U_{\text{imp}}^2})^{\frac{-1}{3}} \tag{2.4}$$

Where:

- K_2 is a material constant,
- a is the impactor radius,
- U_{imp} is the impact velocity.

This scaling relationship is essential in comparing crater formation and ejecta volume across various impact angles **artemievaNumericalSimulationHighvelocity2008a**.

2.3 Ejecta Dynamics and Behavior

2.3.1 Ejecta Velocity and Distribution

Low-gravity environments, such as those on asteroids, have a significant effect on the velocity and distribution of ejecta after an impact. The study **RADUCAN2022114793** investigates how the dynamics of ejecta differ in such conditions. In particular, it was observed that ejecta velocities in low-gravity environments are much lower than on Earth, meaning that ejecta tend to travel slower and remain on the surface for longer periods before settling. The velocity of the ejecta, $v(r)$, from vertical impacts can be described by a modified power law:

$$v(r) = \frac{U}{C1}(\frac{r}{(a)})^{-\mu}(1 - \frac{r}{(n_2)\,R})^p \tag{2.5}$$

Where:

- U is the impact velocity,
- r is the distance from the impact point,
- a is the projectile radius,
- $C1$, p, and $n2$ are material-dependent constants.

This equation helps predict the ejecta speed at different points around the impact zone, illustrating that in low-gravity conditions, particles are more likely to remain gravitationally bound to the asteroid.

2.3.2 Gravity and Escape Velocity Considerations

In low-gravity environments like those on small asteroids, the gravitational force is weak, often resulting in ejecta reaching escape velocities more easily. The weak gravitational pull means that ejecta can travel farther, and some particles might not return to the asteroid surface. The escape velocity v_{esc}, depends on the asteroids mass and radius:

$$v_{esc} = \sqrt{\frac{2GM}{R}} \tag{2.6}$$

Where:

- G is the gravitational constant,
- M is the mass of the asteroid,
- R is the radius of the asteroid.

Given that small asteroids have low escape velocities (on the order of centimeters per second), a significant amount of material may escape during an impact, contributing to the erosion of the asteroid over time **RADUCAN2022114793**.

2.3.3 Non-Spherical Particle Ejecta

Asteroid regolith is rarely composed of perfectly spherical particles. Most of the surface material consists of irregularly shaped particles, which introduces additional complexity in modeling ejecta behavior. In **matuttisUnderstandingDiscreteElement2014** the complexities of non-spherical particle dynamics are discussed Discrete Element Method (DEM) is particularly suited for simulating these granular materials, allowing researchers to model how irregularly shaped particles interact, rotate, and translate during an impact.

- Shape and Rolling Friction: Non-spherical particles exhibit complex interactions like interlocking and increased friction, which reduces the overall velocity of the ejecta. These interactions can be captured using DEM, which models the contact forces between particles:

  $$F_{\text{contact}} = \mu F_n + F_t \tag{2.7}$$

  Where μ is the friction coefficient, F_n is the normal contact force, and F_t is the tangential force.

- Energy Dissipation: Due to their irregular shapes, non-spherical particles tend to dissipate energy more quickly than spherical ones. This leads to slower, more localized ejecta dispersal and may cause particles to settle faster on the asteroid surface.

2.4 Numerical Simulations and Experimental Data

2.4.1 Discrete Element Method (DEM)

The Discrete Element Method (DEM) is essential for simulating the complex interactions between particles in granular materials, such as the regolith on asteroids. DEM models particle collisions, interactions, and forces, making it ideal for studying ejecta dynamics on low-gravity bodies.

- In **ludingIntroductionDiscreteElement2008** Stefan Luding introduces the fundamental aspects of DEM, explaining how it models particle collisions and forces through simplified contact models. DEM treats each particle as an individual body and applies Newton's laws to track their motion, considering forces like gravity, friction, and collisions between particles. In the context of ejecta, DEM enables us to simulate how granular materials are ejected during an asteroid impact and how they interact post-impact.
- Equation for Motion of Particles:

$$m_i \frac{d^2 r_i}{dt^2} = \sum_j f_{ij} + m_i g \tag{2.8}$$

  Where r_i is the position of the particle i, f_{ij} is the contact force between particles i and j and g is the gravitational force.
- Further explaining DEM, in **altenbachEncyclopediaContinuumMechanics2020** emphasizes how DEM models particle collisions and the resulting momentum transfer, a critical aspect of understanding the granular nature of asteroid regolith. DEM models both translational and rotational motion, which is especially important for non-spherical particles.

Modeling Approaches: Lagrangian vs. Eulerian

The fundamental dichotomy in computational particle dynamics manifests through two distinct methodological paradigms: Lagrangian and Eulerian approaches. These frameworks represent fundamentally different philosophical approaches to particle system analysis, as shown in Figure **??**. The Lagrangian approach characterizes discrete elements through individual state vectors incorporating mass properties, positional coordinates, velocity components, and temporal evolution parameters. It demonstrates particular efficacy in micromechanical analyses where particle-level interactions drive system behavior. The computational architecture follows Newtons classical mechanics for particle motion and force-displacement relationships.

Conversely, the Eulerian framework employs a space-fixed reference frame and utilizes continuous field approximations. It implements nodal discretization for field variable computation, conservation equations, and constitutive relationships, thereby facilitating the analysis of bulk behavior through fixed spatial reference points. The selection between these methodologies significantly impacts computational resource allocation, numerical stability considerations, resolution of multi-scale phenomena, and validation against experimental data. This theoretical framework has found particular resonance in diverse applications, from chemical engineering to agricultural processes, demonstrating its versatility in handling complex particulate systems.

Fundamental Principles and Force Mechanisms In DEM

The Discrete Element Method (DEM) is based on Newtons Second Law of Motion, expressed as:

$$F = m \cdot a \tag{2.9}$$

Where F is the net force acting on a particle, m is the mass, and a is the acceleration. This equation is fundamental in simulating the dynamic behavior of particle systems. DEM incorporates force-displacement relationships to model interactions between particles, accounting for both normal and tangential components of contact forces, as shown in Figure **??**. Conservation principles, such as momentum conservation, ensure that the total momentum of the system is conserved unless external forces act upon it. The momentum p of a particle is given by:

$$\vec{p} = m \cdot \vec{v} \tag{2.10}$$

Sphere Contact Classifications

Different models classify sphere contact in DEM. The Hard Sphere Model processes collisions instantaneously with no overlap, making it suitable for dilute systems, as illustrated in Figure **??**. In contrast, the Soft Sphere Model allows minimal overlap, facilitating simultaneous contacts in dense systems. This model is more realistic for packed beds or granular flows, as also depicted in Figure **??** **derekelsworthDiscreteElementMethods2020**.

Force Components Analysis

Contact forces in DEM include normal forces (F_n), which are expressed as:

$$F_n = k_n \cdot \delta_n \tag{2.11}$$

Where k_n is stiffness and δ_n is overlap (Figure **??**). The corresponding tangential force is defined as:

$$F_t = \mu \cdot F_n \tag{2.12}$$

where μ is the friction coefficient. Additionally, non-contact forces such as gravitational force are described by:

$$F_g = G\frac{m_1 m_2}{r_2} \tag{2.13}$$

Where $m1$ and $m2$ represent the mass of the particle, G is the gravitational constant, and r is the distance between the particles. Molecular forces also influence the behavior of contact and non-contact particles, as seen in Figure **??** **derekelsworthDiscreteElementMethods2020**.

Advantages and Applications of DEM

The Discrete Element Method provides significant advantages in micromechanical analysis capabilities, enabling detailed examination of particle-level interactions. It offers granular insights into system behavior and facilitates an understanding of emergent phenomena arising from individual particle dynamics.

DEM has cross-disciplinary applications in various fields. In chemical engineering, it aids in process optimization, particle flow dynamics, and reaction kinetics modeling. The pharmaceutical industry benefits from DEM in powder processing, mixing uniformity analysis, and drug formulation studies. In ceramics manufacturing, DEM assists in material behavior prediction, processing optimization, and quality control modeling. Agricultural engineering leverages DEM for grain transport systems, material separation processes, and storage facility design.

Despite its advantages, DEM comes with certain limitations, including high computational resource requirements, intensive processing demands for large particle systems, and complex validation requirements against empirical data. Practical applications of DEM include concrete mixing optimization, granular material transport, and material separation processes. Additionally, specialized applications such as granular avalanche modeling, transport phenomena analysis, and bulk material handling systems further highlight DEMs versatility across multiple industrial sectors. However, these applications must be balanced with computational constraints and validation requirements to ensure accuracy and efficiency.

2.4.2 Granular Systems in Low Gravity

Granular systems in low-gravity environments, such as on asteroids, behave differently from those in terrestrial conditions. Chapter 6 of Matuttis' textbook, Modeling and Simulation, provides an in-depth explanation of how DEM can be used to model the behavior of granular materials like asteroid regolith under low-gravity conditions **matuttisUnderstandingDiscreteElement2014**. The reduced gravitational force on asteroids affects particle trajectories and ejecta velocities, making it essential to simulate these conditions accurately.

In a low-gravity context, ejecta particles travel farther and remain suspended longer due to the reduced gravitational pull. DEM simulations in this scenario must account for factors like cohesive forces and the low escape velocity of small bodies.

2.4.3 Experimental and Field Studies

In addition to numerical simulations, real-world experiments and field studies provide valuable insights into ejecta behavior on celestial bodies:

- NASA's LCROSS Mission (2015, Advances in Space Research): This mission observed the impact of a spacecraft on the Moon to study the resulting ejecta, offering parallels to asteroid impacts. The LCROSS experiment provided key data on ejecta formation, velocity distribution, and mass, which can be compared to similar impacts on asteroids **LCROSSSatelliteOverview**.
- Hayabusa2 Mission to Ryugu (2021, Science Advances): The Hayabusa2 mission performed an impact experiment using the Small Carry-on Impactor (SCI) to create a crater on the asteroid Ryugu. The experiment collected data on the size and velocity distribution of ejecta, offering valuable comparisons for understanding how regolith behaves in low-gravity environments. The findings from this mission are crucial for validating DEM simulations by providing real-world ejecta behavior **AsteroidExplorerHayabusa2**, **FirstLookRyugu**.

2.5 Knowledge Gaps and Research Opportunities

2.5.1 Challenges in Ejecta Modeling

Modeling the behavior of ejecta for planetary defense scenarios presents numerous challenges. The study **maclennanThermophysicalInvestigationAsteroid2022** reviews the current uncertainties and challenges in modeling ejecta, especially in the context of planetary defense. This paper emphasizes the need for accurate predictions of ejecta behavior to mitigate the risks of potential asteroid impacts. Key issues include the variability in material properties of asteroids, the impact angle, and velocity, which all influence the trajectory and velocity of the ejecta.

- Key Challenges:
  - Material Heterogeneity: The diverse composition of asteroid surfaces complicates modeling efforts. Variations in material properties, such as density and porosity, can lead to different ejecta behaviors.

  - Impact Conditions: The angle and velocity of the impact significantly affect the amount and distribution of ejecta. Oblique impacts tend to produce more asymmetric ejecta patterns compared to vertical impacts.

The DEM in Three Dimensions from Matuttis' textbook discusses the computational challenges of using DEM for large-scale simulations. This chapter highlights the difficulties in simulating realistic ejecta scenarios due to the high computational cost and the complexity of accurately modeling particle interactions in three dimensions **matuttisUnderstandingDiscreteElement2014**.

$$\text{Load} = \mathcal{O}(N \log N) \tag{2.14}$$

Where N is the number of particles, demonstrating the increase in computational resources required for larger simulations.

2.5.2 Future Research Directions

To address these challenges, future research must focus on improving the accuracy and efficiency of ejecta modeling, particularly for small bodies like asteroids.

- Current Limitations and Unresolved Issues: The study **karetaEjectaEvolutionFollowing202** identifies key limitations in current simulations. These include difficulties in replicating the low-gravity environments of asteroids and the complex interactions of non-spherical particles within regolith
- Future Directions:
  - Advanced DEM Techniques: Developing more sophisticated DEM algorithms that can handle larger particle numbers and complex shapes more efficiently.
  - Experimental Validation: Increasing collaboration between experimental and numerical studies to validate DEM models against real-world data from asteroid missions, such as the DART mission and subsequent observations of Dimorphos ejecta.

2.6 Conclusion

2.6.1 Transition to The Study

This research addresses several gaps identified in the literature, particularly the need for more accurate and efficient modeling of ejecta dynamics on asteroids. By applying the Discrete Element Method (DEM), this study aims to:

1. Enhance Ejecta Modeling in Low-Gravity Environments:
    - The research will utilize DEM to simulate the motion of ejecta particles in low-gravity conditions, addressing the unique challenges posed by the weak gravitational pull of asteroids. This includes modeling the slower velocities and extended trajectories of ejecta particles.
2. Incorporate Realistic Particle Shapes:
    - Building on the insights from Matuttis' work, the study will incorporate non-spherical particle dynamics into the simulations. This approach aims to improve the accuracy of ejecta behavior predictions by accounting for the irregular shapes and complex interactions of regolith particles.
3. Validate Models with Experimental Data:
    - The study will compare simulation results with data from missions such as LCROSS and Hayabusa2, ensuring the validity of the models. This validation step is crucial for refining the simulations and improving their predictive capabilities.
4. Identify and Address Computational Challenges:
    - Recognizing the computational difficulties outlined in Chapter 8 of Matuttis' textbook, the study will explore advanced DEM techniques to manage the high computational load and improve the efficiency of large-scale simulations.

By focusing on these aspects, the research will contribute to a deeper understanding of ejecta dynamics on asteroids, enhancing predictive models and informing planetary defense strategies and future asteroid exploration missions.

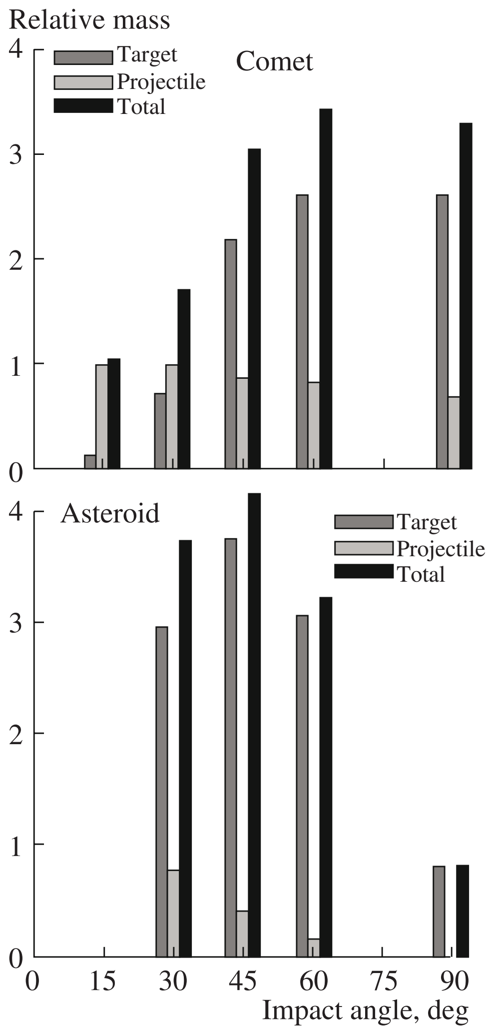


FIGURE 2.4: Compares crater shapes and ejecta patterns between vertical and oblique impacts, highlighting differences in crater symmetry and ejecta reach **artemievaNumericalSimulationHighvelocity2008a**.

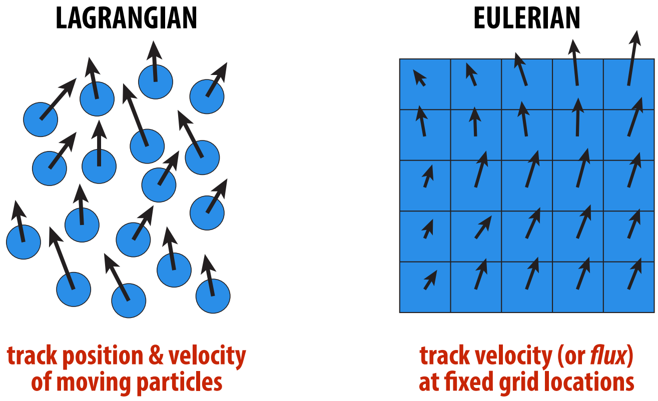


FIGURE 2.5: Illustrate the difference between Lagrangian and Eulerian **SlideViewComputera**.

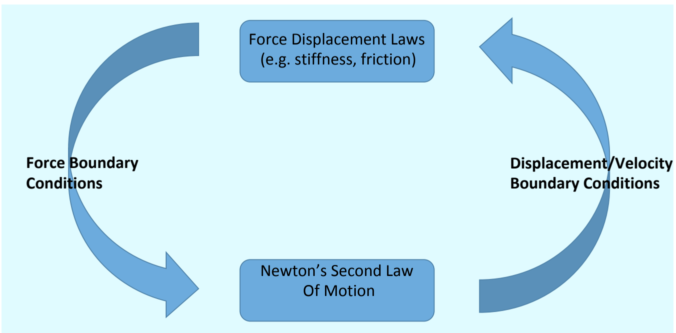


FIGURE 2.6: shows the relationships between force displacement laws and Newton's second law **derekelsworthDiscreteElementMethods2020**.

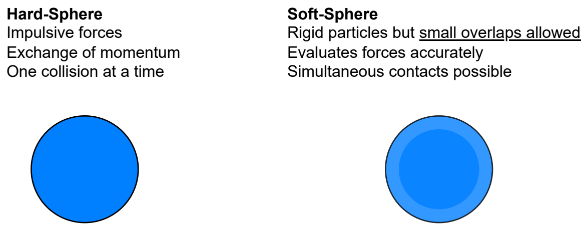


FIGURE 2.7: Shows the difference between the hard sphere and soft sphere **derekelsworthDiscreteElementMethods2020**.

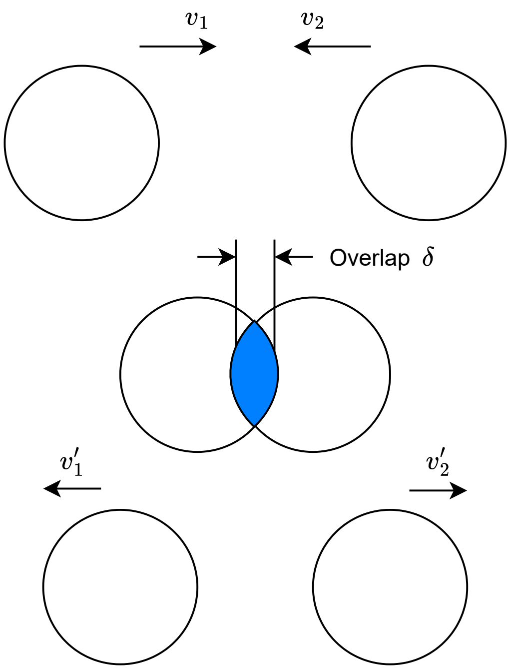


FIGURE 2.8: Illustration how soft spheres overlap happens **derekelsworthDiscreteElementMethods2020**.

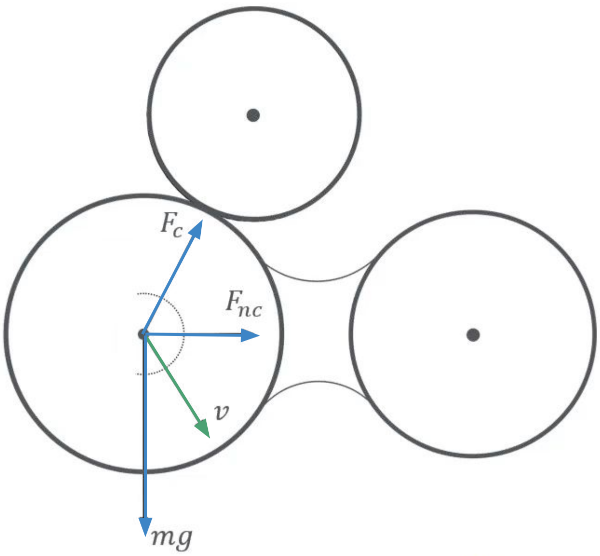


FIGURE 2.9: Forces acting on the particles **derekelsworthDiscreteElementMethods2020**.

CHAPTER 3
RESEARCH METHODOLOGY

3.1 Introduction to YADE

Chapter 3 delves into YADE (Yet Another Dynamic Engine), a powerful open-source platform for simulating discrete element method (DEM) models. Designed with flexibility and precision in mind, YADE has established itself as an indispensable tool for engineers and researchers working on granular materials, multi-body systems, and geomechanical simulations. This introduction covers YADEs history, features, installation, operation, and more.

An example simulation illustrating YADEs capabilities is a gravity deposition scenario. In this setup, particles settle under the influence of gravity to form a stable configuration. This process ensures proper initialization of granular systems, simulating how regolith behaves under specific conditions, such as microgravity see Figure **??**.

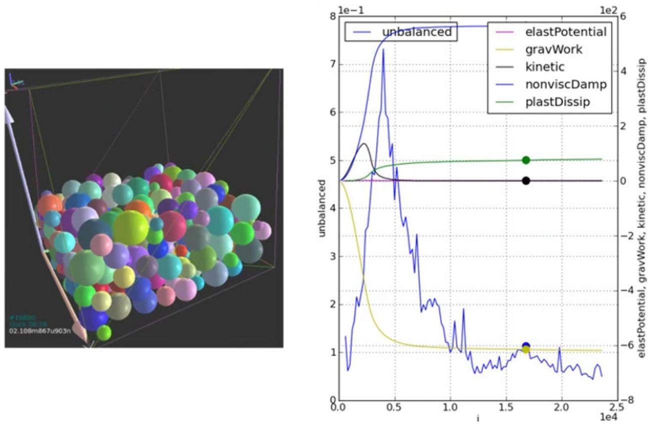


FIGURE 3.1: Gravity deposition example in YADE, demonstrating particle stabilization under gravity **OverviewYade3rd**.

The simulation showcases the deposition process, highlighting how YADE handles granular dynamics efficiently and sets the stage for more complex simulations.

3.1.1 History of YADE

YADE originated as a project from the SDEC group at Grenoble University, designed to meet the growing demand for flexible, extensible DEM software. Over the years, it has evolved through contributions from a global community of developers and researchers. Today, it is hosted on platforms like GitLab and Launchpad and supported by institutions such as the Grenoble Geomechanics Group and Gdask University of Technology **OverviewYade3rd**.

3.1.2 What is YADE?

YADE is an open-source software framework tailored for discrete numerical modeling, primarily using the DEM. It allows users to simulate and analyze the behavior of granular systems and rigid-body interactions under various conditions. With computation-heavy components written in C++ and an intuitive Python interface for scripting, YADE offers both performance and ease of use.

Key features:

- Advanced contact and interaction models for normal, tangential, and rolling forces.
- Support for low-gravity environments, making it ideal for asteroid regolith studies.
- Modular and extensible architecture to incorporate new algorithms and scenarios **OverviewYade3rd**.

3.1.3 How to Install YADE

YADEs compatibility with Ubuntu simplifies its installation and operation for Linux users. Follow these steps to set up YADE on Ubuntu:

1. Update System Packages:

```
sudo apt update && sudo apt upgrade

```

2. Install YADE

```
sudo apt-get install yade

```

3. Run YADE

For Ubuntu users, YADE offers a streamlined installation process and a wealth of support resources from the community **OverviewYade3rd**.

3.1.4 How to Use YADE

YADE provides flexibility through two main interfaces:

1. Python Scripting: Users can write Python scripts to define simulation parameters, set up scenarios, and run simulations. This scripting approach allows for dynamic adjustments and streamlined workflows.
2. Graphical User Interface (GUI): For visualizing and debugging, YADE includes a GUI where users can monitor simulation progress and interact with models in real time.

3.1.5 How YADE Works

YADE can be used in two ways. In this section, I will explain both methods and highlight the preferred one:

1. YADE Terminal: After following the previous section on how to install YADE on Ubuntu, go to the search bar and type *YADE*, as shown in Figure **??**. Once you click on the YADE icon in the search bar, the YADE terminal will open, as shown in Figure **??**. Afterwards, you can write the code for the desired simulation directly in the terminal and run it. However, this method has a significant drawback: saving the YADE code and making corrections or fixing errors can be complicated.
   This is where the second method becomes the preferred choice for most YADE researchers and users:

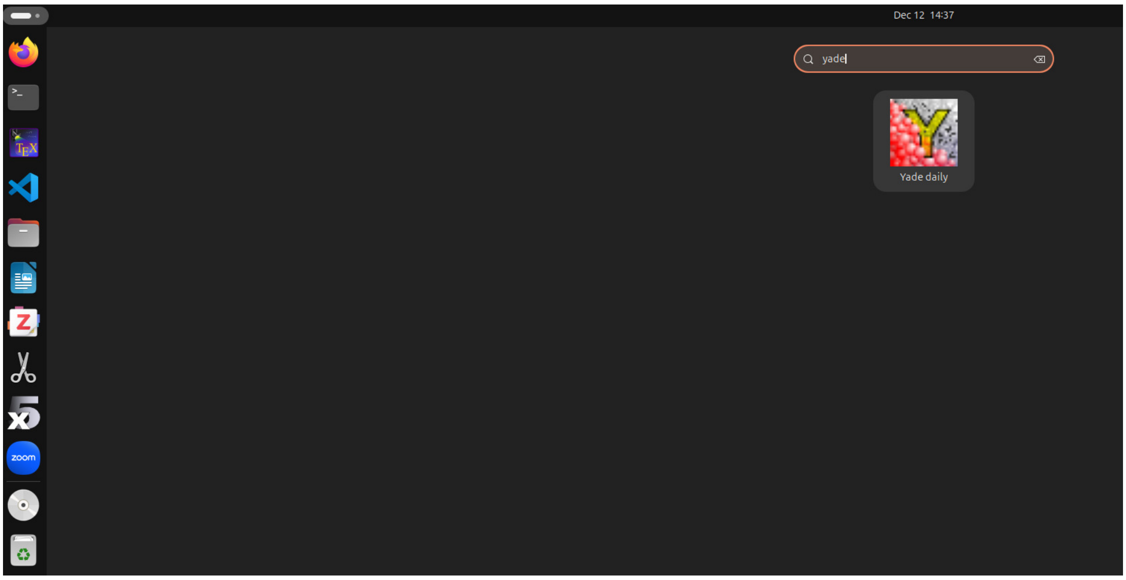


FIGURE 3.2: YADE file icon on Ubuntu.

2. Assigned File for YADE Codings:
   - First, create a designated folder to save all your coding files. This organization makes it easier to apply changes or fix coding errors. For example, I assigned the *Documents* folder for this purpose, as shown in Figure **??**.
   - Next, open the terminal in the assigned folder to run the coding files. To do this, right-click on the *Documents* folder (or the assigned folder) and select *Open in terminal*, as shown in Figure **??**.
   - The terminal for the assigned folder will open. Then, use a text editor to write the code for YADE simulations, as shown in Figures **??** and **??**. Save the code as a Python file in the previously chosen folder, as illustrated in Figure **??**.
   - Finally, return to the terminal and run the file using its saved name, as shown in Figures **??** and **??**.

3.1.6 Why Choose YADE?

YADE stands out as an indispensable tool for researchers, engineers, and scientists who work with simulations of granular systems, multi-body interactions,

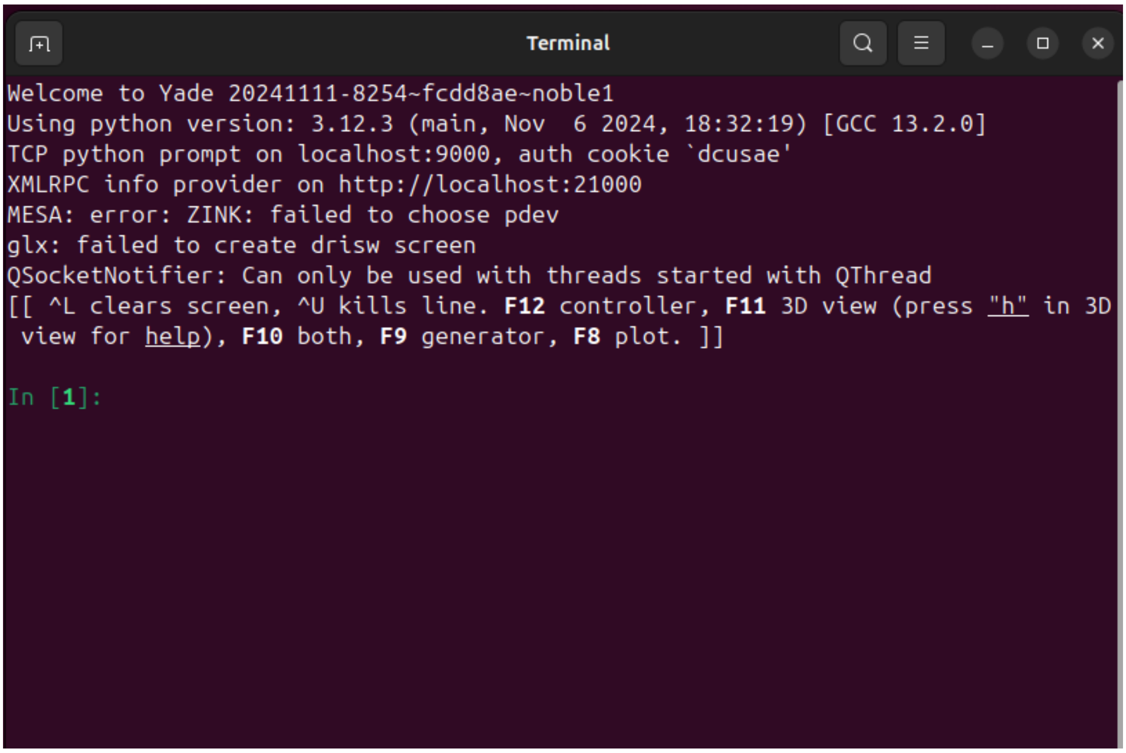


FIGURE 3.3: YADE terminal interface.

and particle dynamics. Its open-source nature, robust architecture, and flexibility make it a premier choice among DEM software. YADE is entirely open-source, making it accessible to a wide audience without the cost barriers of proprietary software. This affordability extends to academic, industrial, and independent users, fostering innovation and collaboration. Additionally, YADEs development is community-driven. Its source code is actively maintained and updated by researchers and developers worldwide. Users benefit from a global network of contributors who provide support, share resources, and develop new features. Continuous updates ensure that YADE remains on the cutting edge of DEM research and applications, and users can directly contribute to or adapt the code to meet unique research needs.

YADEs modular architecture allows users to adapt its functionality to a wide range of problems. Its C++ core ensures high computational performance, while the Python interface offers unmatched flexibility for scripting and customization. The software operates through customizable engines, such as GravityEngine for

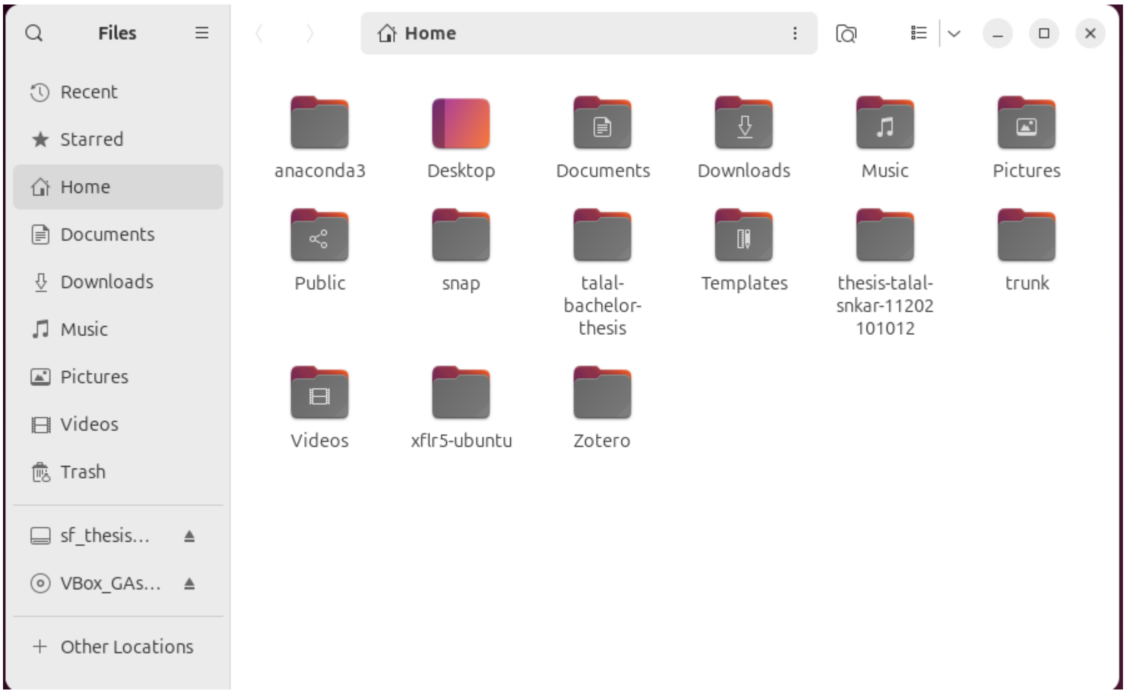


FIGURE 3.4: Assigning *Documents* folder as the destination.

simulating gravitational effects, NewtonIntegrator for solving equations of motion, and various contact models that account for forces like friction and cohesion. Researchers can add new engines or customize existing ones to extend YADEs capabilities for specialized applications. This adaptability makes YADE suitable for diverse scenarios, from planetary science simulations, such as asteroid regolith modeling, to industrial applications like hopper flows or soil mechanics.

YADE offers a comprehensive suite of features for DEM modeling, including support for normal, tangential, rolling, and cohesive forces, enabling accurate modeling of granular materials. It allows users to define material properties such as stiffness, density, and friction, making it highly customizable. YADE is also uniquely capable of simulating environments like asteroid surfaces, making it highly relevant for planetary science research. Additionally, it provides real-time visualization through its graphical user interface (GUI), which is useful for debugging and monitoring simulations. Results can be exported in formats compatible with Paraview and other post-processing tools, further enhancing its usability.

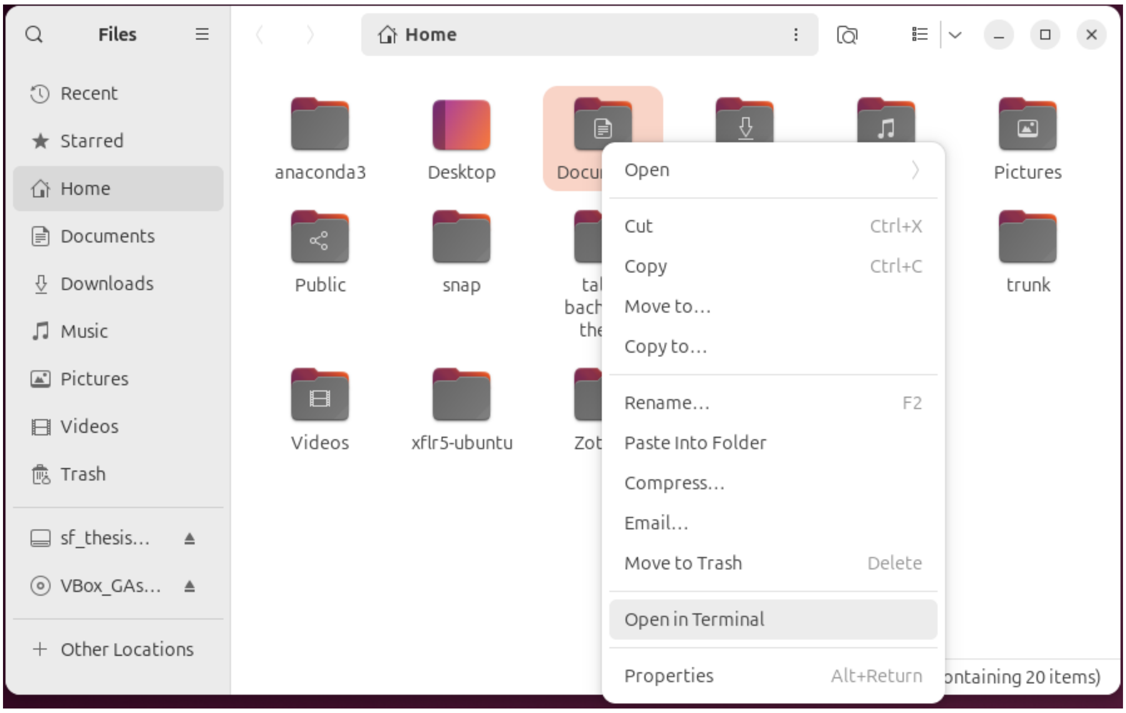


FIGURE 3.5: Opening the terminal in the assigned folder.

Optimized to handle large-scale simulations involving thousands or even millions of particles, YADE utilizes multi-threading to reduce simulation time and efficient algorithms such as bounding volume hierarchies (BVH) for collision detection, minimizing computational overhead. This allows researchers to simulate large systems, such as asteroid ejecta or geotechnical material flows, without sacrificing accuracy or performance. YADE has been validated in various fields, including asteroid studies, geotechnical engineering, and industrial processes. Its extensive documentation, beginner tutorials, and active forums ensure that users have the support they need to start and succeed in their research and projects **OverviewYade3rd**.

3.2 Numerical Model Setup (DEM Simulation)

The numerical model is constructed within the YADE framework using specific parameters that define the simulation environment, particle properties, and boundary conditions. Polyhedral particles are chosen to better simulate the interlocking

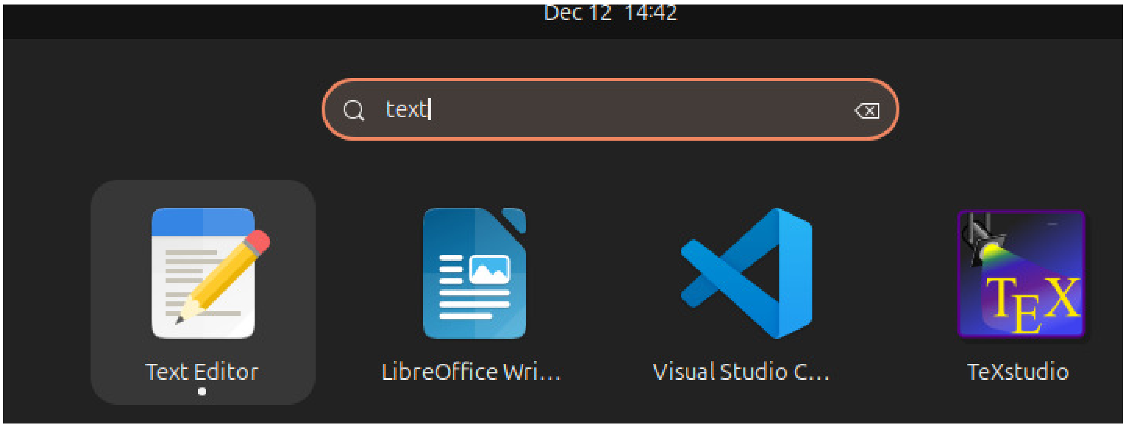


FIGURE 3.6: Using a text editor to write and save the codes.

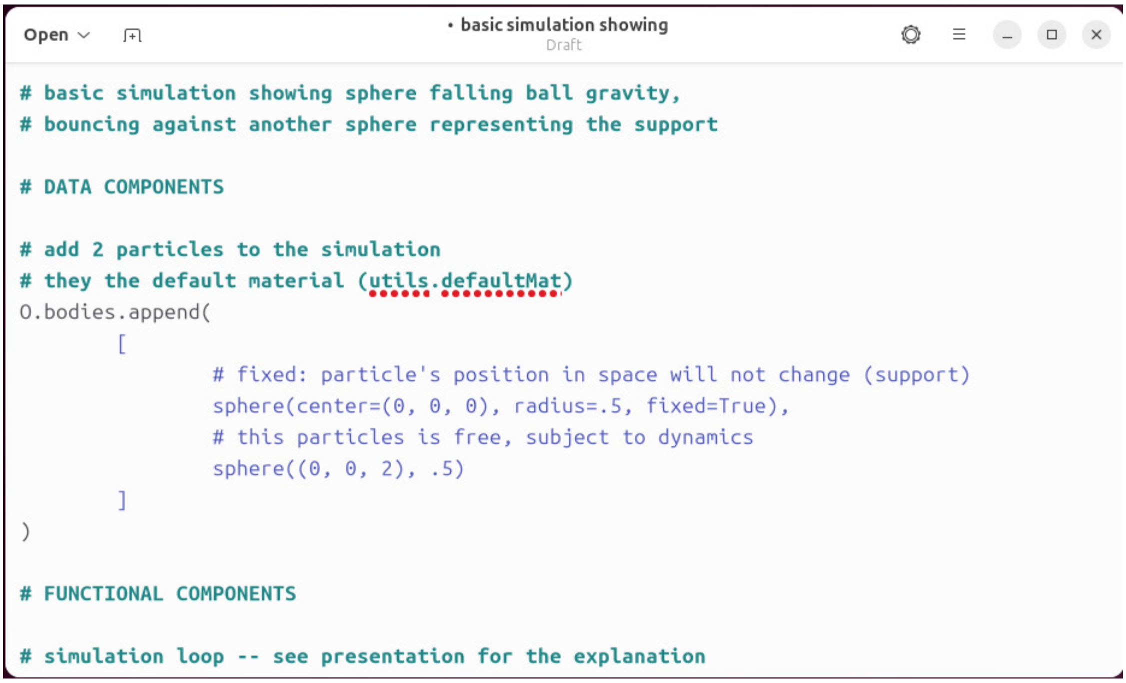
behavior of regolith grains, with a size distribution ranging from 1 to 10 mm. A total of 217 particles are simulated to observe ejecta behavior under an impact velocity of 300 m/s. The impact is modeled as a vertical strike perpendicular to the surface, and a fixed plate represents the asteroid surface to simplify boundary and initial conditions. The DEM approach accurately models the forces acting between particles, including gravitational forces, normal contact forces, tangential frictional forces, and damping.

3.3 Simulation Tools and Software

The simulation is conducted using several computational tools. YADE (Yet Another Dynamic Engine) serves as the core simulation software for discrete element modeling, while Python is employed for pre- and post-processing data, including plotting particle trajectories, energy distributions, and force interactions. Matplotlib is used to visualize force and energy contributions, and YADEs Polyhedra Plugin enables the use of non-spherical particles, which is critical for modeling realistic regolith behavior.

3.3.1 Computational Setup

To ensure reproducibility and provide clarity on computational constraints, the hardware used for running the YADE simulations is specified as follows:




```
# basic simulation showing sphere falling ball gravity,
# bouncing against another sphere representing the support

# DATA COMPONENTS

# add 2 particles to the simulation
# they the default material (utils.defaultMat)
O.bodies.append(
        [
                # fixed: particle's position in space will not change (support)
                sphere(center=(0, 0, 0), radius=.5, fixed=True),
                # this particles is free, subject to dynamics
                sphere((0, 0, 2), .5)
        ]
)

# FUNCTIONAL COMPONENTS

# simulation loop -- see presentation for the explanation
```

FIGURE 3.7: Writing the code in the text editor.

- YADE version: 20241225-8296~a615749~noble1
- Operation system: Virtual box Ubuntu
- System version: Ubuntu 24.04.1 LTS
- RAM provided: 24 514 GB
- Core processor: 6 Cores

3.4 Data Collection and Analysis

Data from the simulations are recorded and analyzed to evaluate particle trajectories, velocity decay, and energy dissipation patterns. Several metrics are considered to interpret the results. Unbalanced force is recorded over time to assess stability during the simulation. Energy contributions are categorized into kinetic energy, which experiences initial spikes due to high-velocity impacts, elastic potential energy from particle deformation, gravitational work influenced by microgravity, and dissipative energy resulting from non-viscous damping effects.

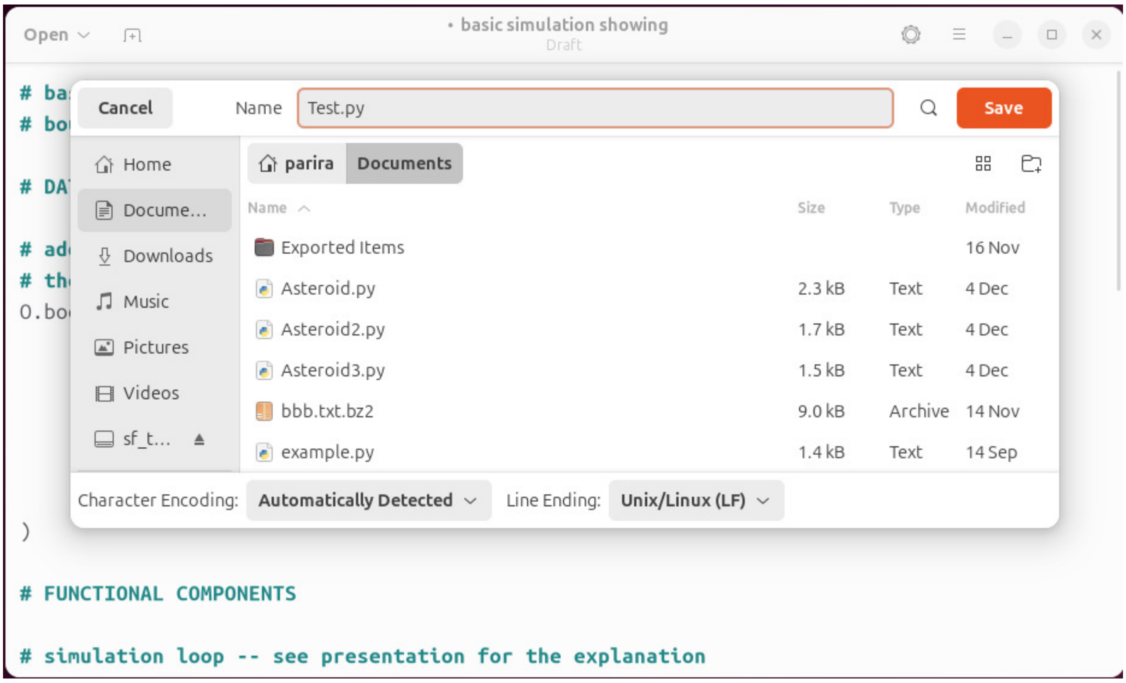


FIGURE 3.8: Saving the coding file as a Python file.

The analysis involves plotting energy components against time to observe how kinetic energy transitions into other forms, highlighting energy loss patterns typical of microgravity environments. Results are visualized through force and energy balance diagrams, revealing characteristic decay and peak patterns that define ejecta dynamics on asteroid surfaces.

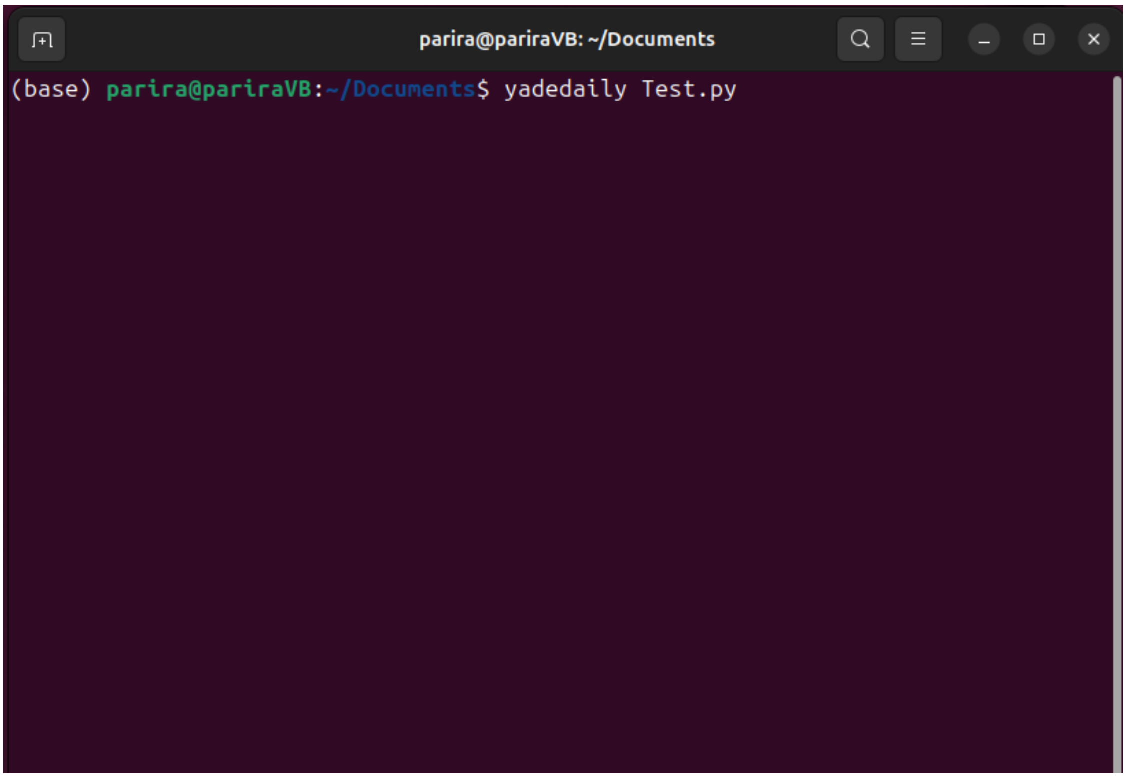


FIGURE 3.9: Running the YADE code file.

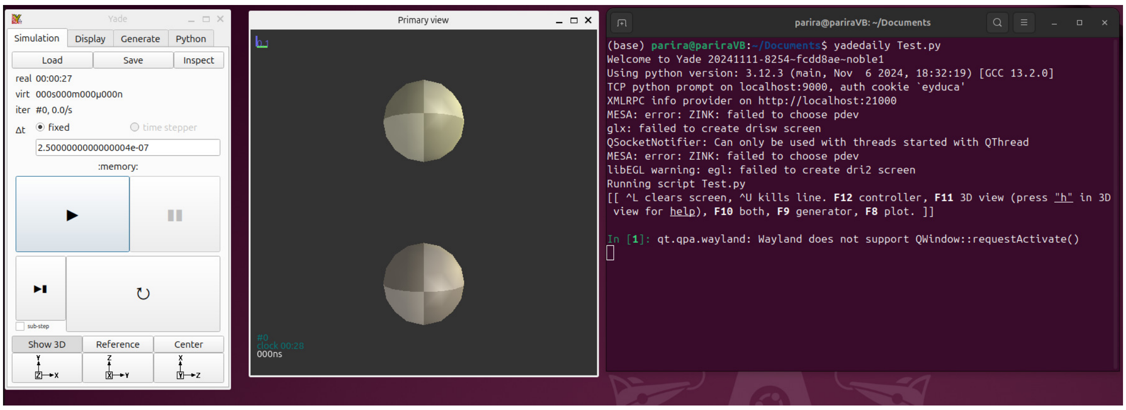


FIGURE 3.10: Simulation output of the coding file.

CHAPTER 4
RESULTS AND DISCUSSIONS

4.1 Simulation Results

The simulation of ejecta behavior on an asteroid-like surface was conducted using YADE, a discrete element method (DEM) tool, to analyze particle interactions under conditions mimicking the asteroid Ryugu. The simulation setup consisted of a flat plate representing the asteroid surface with Ryugu-like properties see Table **??**. The regolith was modeled using 217 polyhedral particles ranging in size from $1\ \mathrm{mm}$ to $10\ \mathrm{mm}$, and an impacting sphere with a radius of $8.22\ \mathrm{mm}$ was ejected downward at a velocity of $300\ \mathrm{m/s}$ see Table **??**.

In the simulation we're using an ejected sphere from $1\ \mathrm{m}$ height with a properties of Tantalum alloy with a downward speed of $300\ \mathrm{m/s}$ see Table **??** for the ejected sphere properties

Parameter	Value
Surface Diameter	$100\ \mathrm{mm}$
Radius of The Surface	$50\ \mathrm{mm}$
Gravity	$0.00011\ \mathrm{m/s^2}$
Density	$1200\ \mathrm{kg/m^3}$
Young Modules	$7.4\ \mathrm{GPa}$
Poisson	0.25
Friction Angle	$30\ \mathrm{Rad}$

TABLE 4.1: Illustrates the surface configuration.

To make a simulation of Ryugu asteroid ejecta behavior choosing an irregular particles was the best option to achieve the most accurate results. Using Polyhedra with 217 number of particles, see the Table **??** for the static particles configuration.

Parameter	Value
Radius	8.22 mm
Density	16678 kg/m^3
Young Modules	186 GPa
Poisson	0.34
Mass	4.85 g

TABLE 4.2: Illustrates the ejected sphere configuration **Tantalum2025**.

Parameter	Value
D_{min}	1 mm
D_{max}	10 mm
Density	1282 kg/m^3
Young Modules	7.223 GPa
Poisson	0.3
Friction Angle	0.6 Rad

TABLE 4.3: Illustrates the static particles configuration.

The results of the DEM simulations are visualized across three key stages to better understand regolith dynamics during an asteroid impact event:

1. Before Impact: The initial configuration of regolith particles shows them settled and stabilized on a fixed plate surface. Gravity acts uniformly across the system, creating a baseline for the impact simulation see Figure **??**.
2. During Impact: This stage captures the moment of projectile impact at 300 $\mathrm{m/s}$. Kinetic energy is transferred from the projectile to the regolith, displacing particles outward and forming ejecta see Figure **??**.
3. After Impact: Post-impact, the particles exhibit a spread outward with some reaching escape velocity while others resettle on the surface. Energy dissipation is apparent in the reduced velocities of the particles see Figure **??**.

Key simulation parameters were defined to accurately mimic the weak gravity conditions of small asteroids. The gravitational acceleration was set at 0.00011 $\mathrm{m/s^2}$, while material properties incorporated a friction angle of 30 degrees to

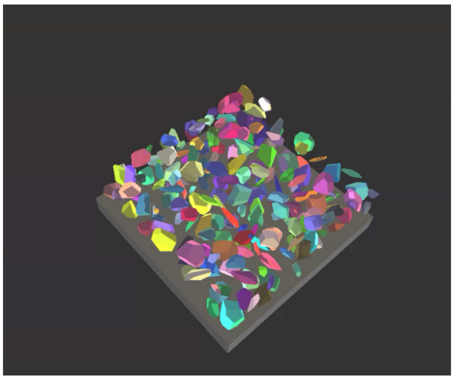


FIGURE 4.1: Initial configuration of regolith particles before the impact. Particles are evenly distributed and stabilized, representing an asteroid's surface.

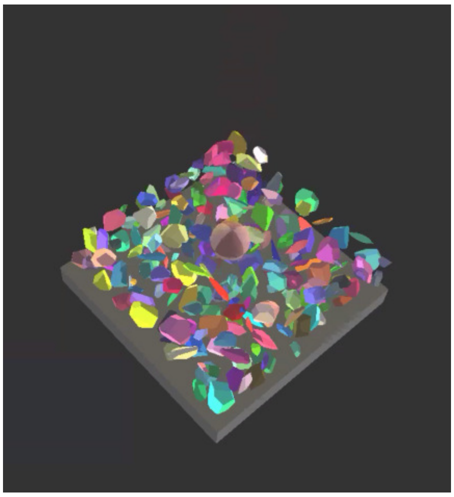


FIGURE 4.2: Visualization of regolith particle motion during the impact event. Energy transfer and initial particle ejection are evident.

simulate regolith cohesion and shear resistance. Visual outputs from the simulations illustrated the particle behavior at different stages, including the initial configuration of polyhedral particles resting on a planar surface, the impact event characterized by a high-velocity sphere strike causing a burst-like dispersal of particles, and the resulting ejecta trajectories showing both upward and radial motions that varied by distance from the impact center. To further analyze ejecta behavior, additional simulations were performed using spherical particles instead of polyhedral ones. These simulations covered three different sphere size distributions: uniform $1\ \mathrm{mm}$ radius spheres, variable radius spheres ranging from 1

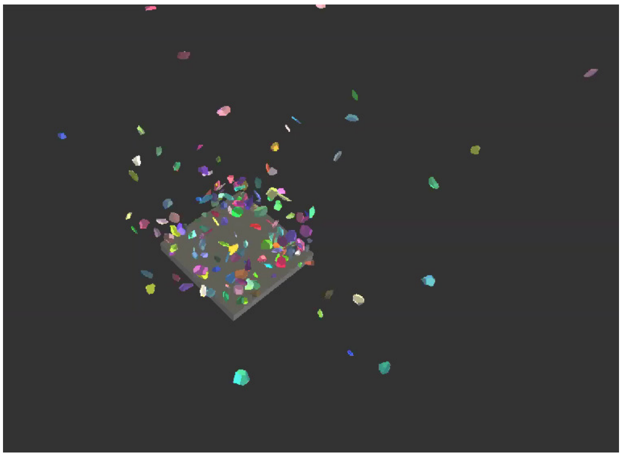


FIGURE 4.3: Final distribution of particles after the impact, showing ejecta dispersal and stabilization.

mm to $10\ \mathrm{mm}$, and uniform $10\ \mathrm{mm}$ radius spheres. Each case was tested with impactor velocities of $250\ \mathrm{m/s}$, $300\ \mathrm{m/s}$, and $350\ \mathrm{m/s}$. The key output variables analyzed included maximum velocity of particles post-impact ($v_{\max}$), the number of escaped particles, and the overall ejecta motion characteristics, including bouncing, stabilization, and escape patterns.

The following Table **??** summarizes the results:

4.2 Analysis of Ejecta Motion

The motion of ejecta particles was analyzed to understand dispersal patterns under low-gravity conditions. Velocity distribution analysis indicated that ejecta velocity exhibited a sharp gradient, with the highest velocities near the impact point. The relation between velocity and radial distance followed a modified power-law equation:

$$v_{(r)} = v_0 (\frac{r}{R})^{-\alpha} \tag{4.1}$$

where $v_{(r)}$ represents the velocity at distance r, v_0 is the initial ejection velocity, R is the maximum radial distance of ejecta, and α is a material-dependent constant. The results demonstrated that ejecta velocities decreased with increasing distance from the impact center, aligning with energy dissipation trends and

Simulation	Particle Radius mm	Impactor Velocity m/s	$v_{\max}$ (m/s)	Generated Particles	Escaped Particles
1	1	350	246	217	217
2	1	300	175	217	217
3	1	250	146	217	217
4	1 - 10	350	246	50	5
5	1 - 10	300	175	50	14
6	1 - 10	250	146	50	11
7	10	350	246	15	Bounced 2-3 times, then escaped
8	10	300	175	15	7 escaped immediately, others bounced multiple times before escaping
9	10	250	146	15	3 escaped immediately, others bounced before eventually escaping

TABLE 4.4: Summary of Ejecta Behavior for Different Particle Sizes and Impact Velocities.

gravitational influence. Parabolic trajectories were observed, characteristic of microgravity environments. Reduced gravitational pull allowed particles to travel longer distances before settling, with simulations capturing both primary ejecta motion driven directly by impact and secondary collisions contributing to further dispersion. Impact energy dissipation was evident as kinetic energy peaked during the initial ejection and subsequently declined as particles lost momentum due to frictional and collisional forces.

Parameter	Value
Ejected Sphere Radius	8.22 mm
Sphere Impact Speed	300 m/s
Gravity	0.00011 m/s^2
Particle Count	217
Particle Size Range	$1 - 10$ mm

TABLE 4.5: Illustrates the initial particle configuration.

4.2.1 Effect of Particle Size on Ejecta Behavior

Simulations revealed distinct trends based on particle size. Small particles with a $1\ \mathrm{mm}$ radius nearly all reached escape velocity regardless of impactor speed due to low inertia and minimal interparticle collisions, facilitating efficient energy transfer. The velocity distribution remained uniform, with Vmax increasing proportionally with impact velocity. In contrast, mixed-size particle simulations ($1\ \mathrm{mm}$ to $10\ \mathrm{mm}$ radius) showed that larger particles were significantly less likely to escape, leading to greater variation in ejecta behavior. Some particles rebounded off the surface while others settled, and at $350\ \mathrm{m/s}$, fewer particles escaped than at $300\ \mathrm{m/s}$ or $250\ \mathrm{m/s}$, suggesting that higher impact energy led to increased interparticle collisions, dissipating energy before complete escape. Large particles ($10\ \mathrm{mm}$ radius) required multiple rebounds before escaping. $250\ \mathrm{m/s}$, only three particles escaped immediately, while the rest bounced before eventual ejection. At $300\ \mathrm{m/s}$, ejecta behavior was more diverse, with some particles continuing to bounce for an extended period before escaping.

4.2.2 Effect of Impact Velocity on Ejecta Behavior

Higher impact velocities consistently resulted in greater $v_{\max}$ but did not always correlate with an increase in escaped particles. At $250\ \mathrm{m/s}$, particles exhibited more bouncing before escaping, whereas at higher velocities, escape was more immediate. Increased impact velocity enhanced energy transfer but also promoted greater interparticle interactions, leading to energy dissipation, particularly in mixed-size and large-particle simulations.

4.2.3 Observations on Energy Dissipation

- Small uniform particles experienced minimal energy dissipation, resulting in near-total escape.
- Larger particles experienced greater energy dissipation due to interparticle collisions and surface interactions, leading to prolonged bouncing before escape.

- Mixed-size simulations demonstrated a transition between efficient escape and energy-dissipative collisions, making them more representative of real asteroid regolith behavior.

4.3 Energy Dissipation Dynamics

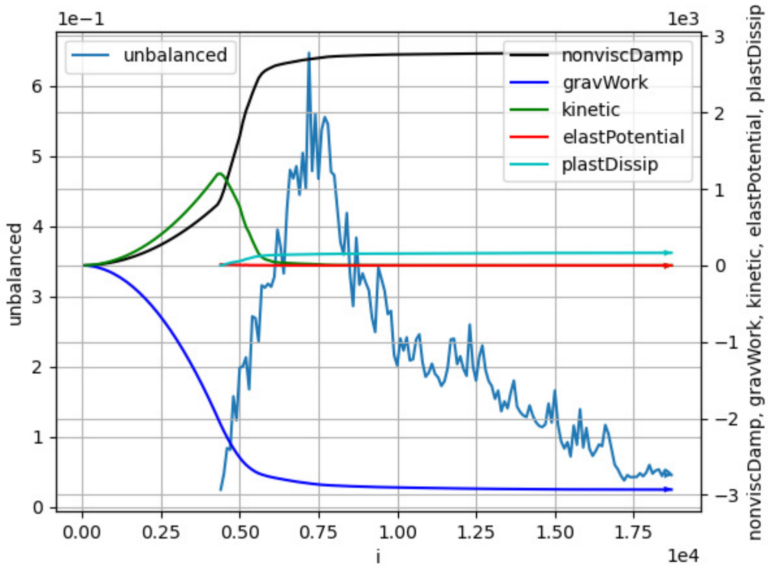


FIGURE 4.4: Evolution of energy dissipation and unbalanced force during regolith simulation on an asteroid surface. The plot shows kinetic energy (green), elastic potential energy (red), gravitational work (blue), non-viscous damping (black), plastic dissipation (cyan), and unbalanced force (light blue).

The evolution of key energy parameters was analyzed throughout the simulation. The unbalanced force curve indicated system stability, with an initial sharp decrease followed by stabilization. Kinetic energy peaked at the moment of projectile impact and then declined as energy dissipated through frictional and collisional interactions. Gravitational work remained constant, reflecting the negligible effect of gravity in microgravity conditions. Elastic potential energy remained nearly constant, indicating minimal elastic recovery, while plastic dissipation increased post-impact, highlighting permanent particle deformation and rearrangement. The role of non-viscous damping was also observed, reducing oscillatory motion and contributing to overall regolith stabilization as seen in Figure **??**, and Table **??**.

Energy/Force	Observed Behavior	Notes on Simulation Outcome
Kinetic Energy	Initial sharp increase due to impact force	Peak occurs immediately following impact
Gravitational Work	Minor contribution in microgravity	Gradual, consistent influence on particle motion
Elastic Energy	Fluctuates during particle deformation	Critical in non-spherical interactions
Dissipative Energy	Steady increase indicating energy loss	Dominated by contact friction and particle interactions

TABLE 4.6: Energy Contributions.

4.3.1 Relevance to Research Objectives

This analysis directly addresses Objective (analyzing the influence of particle geometry and microgravity on ejecta dynamics) by demonstrating how polyhedral particle models enhance energy dissipation realism. Unlike spherical models, which often overestimate kinetic retention due to reduced interlocking, the polyhedral approach provides more accurate frictional and plastic dissipation effects.

4.4 Implications for Asteroid Studies

The findings from these simulations offer significant contributions to asteroid exploration. The motion of particles in a low-gravity environment reveals fundamental differences from Earth-based impact dynamics, highlighting prolonged airborne trajectories and broader dispersal ranges. The simulations demonstrate that smaller particles are more prone to escaping, leading to long-term size segregation on asteroid surfaces. Repeated impact events contribute to regolith redistribution, influencing crater formation, slope stability, and surface texture evolution. Larger grains tend to remain on the surface longer, while finer particles are ejected more easily, providing insights into observed regolith distributions on asteroids such as Ryugu and Bennu. These findings also inform spacecraft landing strategies by improving predictions of ejecta behavior, helping mitigate risks posed by high-velocity debris. Additionally, the study provides valuable data for planetary defense missions by refining kinetic impactor models for asteroid deflection strategies.

4.5 Limitations

Despite the valuable insights, the study has several limitations. The simulation used polyhedral particles, but real regolith features more irregular, non-convex shapes. The model considered a single impact event, whereas real asteroid surfaces experience cumulative bombardments. Environmental factors such as solar radiation and electrostatic forces were excluded. Additionally, computational constraints limited the complexity and scale of the simulations. Running large-scale simulations with highly detailed physics requires significant computational power, which can be cost-prohibitive. Future studies could explore higher-resolution simulations with more particles to enhance accuracy and predictive capabilities.

CHAPTER 5
SUMMARY, CONCLUSION, RECOMMENDATION

5.1 Summary

This thesis explored multiple aspects of ejecta dynamics on asteroid surfaces using discrete element method (DEM) simulations. The study first examined the physical properties of small rubble-pile asteroids such as Itokawa and Ryugu, emphasizing their unique regolith characteristics, including variations in grain size, density, and composition, which influence ejecta behavior. The power-law relationship governing particle size distribution and the weak gravitational environment of these asteroids allow ejecta particles to travel significant distances. The suitability of the open-source YADE platform for DEM-based asteroid surface modeling was demonstrated, with a focus on its modular engines, Python integration, and customizable material properties. The platforms flexibility in particle shape modeling and ability to handle large-scale simulations were highlighted, along with its computational cost and complexity limitations.

The research involved numerical simulations of polyhedral particle regolith surfaces subjected to high-velocity sphere impacts. The analysis included velocity decay trends following a power-law distribution, parabolic trajectories under microgravity conditions, and particle interactions resulting in secondary scattering. Energy balance validation was performed through kinetic energy peaks during impact and subsequent energy dissipation mechanisms. The study provided a comprehensive framework for understanding ejecta motion under low-gravity conditions and its broader implications for planetary science.

5.2 Conclusion

This research successfully simulated and analyzed ejecta motion on an asteroid-like surface using the Discrete Element Method (DEM) in YADE, addressing the research objectives systematically. The first objective, which aimed to simulate ejecta motion using DEM with polyhedral particle modeling, was achieved by developing a numerical model that accurately represents the interactions between irregular regolith particles under microgravity conditions. The use of polyhedral particles enhanced the realism of particle-particle interactions compared to traditional spherical approximations, allowing for better representation of interlocking behaviors observed in asteroid regolith.

The second objective, investigating how impactor speed influences ejecta dynamics, was addressed by conducting simulations across different impact velocities. The results demonstrated that higher impact velocities generally resulted in greater maximum ejecta velocities ($v_{\max}$) but did not always lead to an increased number of escaped particles. Instead, energy dissipation through interparticle collisions played a significant role in determining final ejecta motion. The study established that ejecta velocity patterns decrease with radial distance following a well-defined power-law model, consistent with theoretical models of impact cratering in low-gravity environments.

The third objective, studying ejecta behavior under conditions similar to actual asteroid operations, was met through detailed analysis of particle motion, energy dissipation, and velocity decay patterns. The results confirmed that microgravity significantly extends ejecta flight times and dispersal distances, with smaller particles exhibiting higher escape rates than larger ones. These findings align with existing studies on asteroid regolith mobility and crater evolution. Additionally, the study successfully modeled energy dynamics on an asteroid, providing insights into regolith responses to impacts.

The fourth objective, addressing complexities in designing sample collection mechanisms, was partially fulfilled by demonstrating how different particle shapes and impact conditions affect ejecta dispersal. The insights gained from these simulations have direct implications for mission planning, particularly in optimizing sample retrieval techniques and ensuring spacecraft safety during landing operations. Understanding ejecta motion in low-gravity environments is essential for refining sampling strategies and mitigating risks associated with regolith disturbances.

Overall, this research demonstrated the effectiveness of DEM simulations using YADE to replicate ejecta behavior on asteroid-like surfaces. These findings contribute to more accurate predictions of surface behavior, benefiting the design of future sampling missions and spacecraft operations. Future work should expand on multi-impact scenarios and integrate additional external forces, such as solar radiation pressure and electrostatic interactions, to further enhance the models fidelity.

5.3 Recommendations

Several improvements can enhance the accuracy and applicability of future research in this domain. First, implementing more complex particle shapes, such as non-convex polyhedral or multifaceted models, would better replicate natural regolith behavior. Additionally, simulating multiple impact events rather than a single impact would provide insights into the cumulative evolution of asteroid surfaces and crater chain formation. Incorporating environmental forces such as solar radiation pressure, cohesion effects, and electrostatic interactions could improve the fidelity of ejecta modeling.

Further studies could explore varying impact angles and projectile shapes to expand understanding of regolith behavior under diverse impact conditions. Computational optimization through parallel computing or GPU-based approaches would significantly reduce simulation times, increasing the feasibility of larger-scale studies. Validation of simulation results through direct comparison with experimental data from missions such as Hayabusa2 and OSIRIS-REx would strengthen model reliability. Finally, expanding these studies to different asteroid types, such as S-type and C-type asteroids, would broaden the applicability of findings to a diverse range of surface compositions, aiding both planetary exploration and asteroid impact mitigation strategies. Addressing these areas will refine DEM simulations for planetary science, contributing to exploration, defense strategies, and a deeper understanding of asteroid surface processes.

Appendices

Appendix A: YADE Python Codes

```
from yade import pack, polyhedra_utils, geom, utils, qt, export,
  plot
import math

global sphereAdded
sphereAdded = False

# Parameters
g = 0.00011  # Gravity (m/s^2)
d_plate = .1 #Diameter of the flat plate (m)
r_plate = d_plate / 2  # Radius of the flat plate
thickness = 0.01 # Thickness of the plate (m)

# Material for the polyhedral particles
poly_mat = PolyhedraMat()
poly_mat.density = 1282  # kg/m^3
poly_mat.young = 7.233E9  # Pa
poly_mat.poisson = 0.3
poly_mat.frictionAngle = 0.6  # rad

plate_mat = FrictMat(young=7.4e9, poisson=0.25, frictionAngle=
 math.radians(30), density=1200)

# Create the flat plate
O.bodies.append(
polyhedra_utils.polyhedra(
poly_mat,
v=((-r_plate , -r_plate , -thickness / 2), (r_plate , -r_plate ,
  -thickness / 2), (r_plate , r_plate , -thickness / 2), (-
 r_plate , r_plate , -thickness / 2), (-r_plate , -r_plate ,
 thickness / 2), (r_plate , -r_plate , thickness / 2), (r_plate
 , r_plate , thickness / 2), (-r_plate , r_plate , thickness /
 2)),
```

```
fixed=True,
color=(0.35, 0.35, 0.35),

)
)



# Create a pack of smaller polyhedral particles on the plate
  surface
# Adjusted box dimensions for polyhedra packing
pack_extent = 1.0  # Extent of the packed polyhedra (m)
polyhedra_pack = polyhedra_utils.fillBox(
(-r_plate , -r_plate , thickness ),  # Bottom aligned with the
  plate surface
(r_plate , r_plate, thickness + 0.02),
material=poly_mat,
sizemin=(0.001, 0.001, 0.001),
sizemax=(0.01, 0.01, 0.01),
seed=32
)

if polyhedra_pack is not None:
for p in polyhedra_pack:
O.bodies.append(p)
else:
print("No polyhedra generated. Check box dimensions and size
  constraints.")

# Controller to add a sphere after 20 seconds
sphereAdded = False

def controller(generateSphereAt=1):#sphere will be generated
  after one second.
global sphereAdded
if not sphereAdded and O.time > generateSphereAt:  # Add sphere
  after a certain time
sphereAdded = True
frictMat = FrictMat(young=186E9, density=16678, poisson=.34)
sph = sphere((0.0, 0.0, 1.0), .00822, material=frictMat)
```

```
O.bodies.append(sph)
sph.state.vel = (0, 0, -300)

# Simulation engines
O.engines = [
ForceResetter(),
InsertionSortCollider([Bo1_Polyhedra_Aabb(), Bo1_Facet_Aabb(), Bo1_Sphere_Aabb()]),
InteractionLoop(
[Ig2_Facet_Polyhedra_PolyhedraGeom(),
Ig2_Polyhedra_Polyhedra_PolyhedraGeom(), Ig2_Sphere_Polyhedra_ScGeom()],
[Ip2_PolyhedraMat_PolyhedraMat_PolyhedraPhys(), Ip2_FrictMat_FrictMat_FrictPhys()],
[Law2_PolyhedraGeom_PolyhedraPhys_Volumetric(), Law2_ScGeom_FrictPhys_CundallStrack()]
),
NewtonIntegrator(gravity=(0, 0, -g), damping=0.5),
PyRunner(command='checkUnbalanced()', realPeriod=2),  # Call the checkUnbalanced function every 2 seconds
PyRunner(command='addPlotData()', iterPeriod=100),    # Call the addPlotData function every 200 steps
PyRunner(command='controller(generateSphereAt=1)', virtPeriod=0.1)
]

O.trackEnergy = True

def checkUnbalanced():
if unbalancedForce() < .05:
plot.saveDataTxt('simulation_data.txt.bz2')

def addPlotData():
plot.addData(i=O.iter, unbalanced=unbalancedForce(), **O.energy)

# Time step
O.dt = 0.5 * PWaveTimeStep()

plot.plots = {'i': ('unbalanced', None, O.energy.keys)}

```

```
# Show the plot on the screen, and update while the simulation
 runs
plot.plot()

# Save the simulation state
O.saveTmp()
```

LISTING 1: Polyhedral Simulation YADE Code

```
from yade import pack, polyhedra_utils, geom, utils, qt, export,
  plot
import math

global sphereAdded
sphereAdded = False


# Parameters
g = 0.00011  # Gravity (m/s^2)
d_plate = .1 #Diameter of the flat plate (m)
r_plate = d_plate / 2  # Radius of the flat plate
thickness = 0.01 # Thickness of the plate (m)

# Material for the polyhedral particles
poly_mat = PolyhedraMat(density=1282, young=7.233E9, poisson
 =0.3, frictionAngle=0.6)
plate_mat = FrictMat(young=7.4e9, poisson=0.25, frictionAngle=
 math.radians(30), density=1200)

O.materials.append(poly_mat)
O.materials.append(plate_mat)

# Create the flat plate as a polyhedra
O.bodies.append(
polyhedra_utils.polyhedra(
poly_mat,
v=((-r_plate, -r_plate, -thickness / 2), (r_plate, -r_plate, -
 thickness / 2),
(r_plate, r_plate, -thickness / 2), (-r_plate, r_plate, -
 thickness / 2),
```

```
(-r_plate, -r_plate, thickness / 2), (r_plate, -r_plate,
 thickness / 2),
(r_plate, r_plate, thickness / 2), (-r_plate, r_plate, thickness
 / 2)),
fixed=True,
color=(0.35, 0.35, 0.35),
)
)

# Create a pack of exactly 217 spherical particles on the plate
 surface
sphere_pack = pack.SpherePack()
sphere_pack.makeCloud(
minCorner=(-0.015, -0.015, 0.005),
maxCorner=(0.015, 0.015, 0.005 + 0.005),
rMean=0.001,  # Mean radius of 1 mm
num=217,  # Specify the exact number of spheres
seed=32
)

# Add the spheres to the simulation
for center, radius in sphere_pack:
O.bodies.append(utils.sphere(center, radius, material=poly_mat))

# Controller to add a sphere after 20 seconds
sphereAdded = False

def controller(generateSphereAt=1):  # Sphere will be generated
 after one second.
global sphereAdded
if not sphereAdded and O.time > generateSphereAt:  # Add sphere
 after a certain time
sphereAdded = True
frictMat = FrictMat(young=186E9, density=16678, poisson=0.34)
O.materials.append(frictMat)
sph = utils.sphere((0.0, 0.0, 1.0), .00822, material=frictMat)
O.bodies.append(sph)
sph.state.vel = (0, 0, -250)

# Add the functions to track Vmax and escaped particles
```

```
escapedParticles = 0  # Counter for particles that escape
def trackVelocityAndEscaped():
global escapedParticles
maxVelocity = 0  # To find Vmax
escaped = 0  # Temporary counter for escaped particles this iteration

for b in O.bodies:
if isinstance(b.shape, Sphere):  # Only track spheres
vel = b.state.vel.norm()  # Magnitude of velocity
maxVelocity = max(maxVelocity, vel)

# Check if the particle has flown out of bounds
if b.state.pos[2] > 0.05 or b.state.pos[2] < -0.02:  # Adjusted escape condition
escaped += 1

escapedParticles += escaped  # Add to total count
plot.addData(
i=O.iter,
Vmax=maxVelocity,
escapedParticles=escapedParticles

)

# Simulation engines
O.engines = [
ForceResetter(),
InsertionSortCollider([Bo1_Polyhedra_Aabb(), Bo1_Facet_Aabb(), Bo1_Sphere_Aabb()]),
InteractionLoop(
[Ig2_Facet_Polyhedra_PolyhedraGeom(),
Ig2_Polyhedra_Polyhedra_PolyhedraGeom(), Ig2_Sphere_Polyhedra_ScGeom()],
[Ip2_PolyhedraMat_PolyhedraMat_PolyhedraPhys(), Ip2_FrictMat_FrictMat_FrictPhys()],
[Law2_PolyhedraGeom_PolyhedraPhys_Volumetric(), Law2_ScGeom_FrictPhys_CundallStrack()]
),
NewtonIntegrator(gravity=(0, 0, -g), damping=0.5),
```

```
PyRunner(command='controller(generateSphereAt=1)', virtPeriod
  =0.1),
PyRunner(command='trackVelocityAndEscaped()', iterPeriod=100)
]

O.trackEnergy = True
O.dt = 0.5 * PWaveTimeStep()

plot.plots = {'i': ('Vmax', 'EscapedParticle')}
plot.plot()

# Save the simulation state
O.saveTmp()
```

LISTING 2: Sphere YADE Simulation Code 1 mm

```
from yade import pack, polyhedra_utils, geom, utils, qt, export,
    plot
import math
import random

global sphereAdded
sphereAdded = False

# Parameters
g = 0.00011  # Gravity (m/s^2)
d_plate = .1 #Diameter of the flat plate (m)
r_plate = d_plate / 2  # Radius of the flat plate
thickness = 0.01 # Thickness of the plate (m)

# Material for the polyhedral particles
poly_mat = PolyhedraMat(density=1282, young=7.233E9, poisson
  =0.3, frictionAngle=0.6)
plate_mat = FrictMat(young=7.4e9, poisson=0.25, frictionAngle=
  math.radians(30), density=1200)

O.materials.append(poly_mat)
O.materials.append(plate_mat)

# Create the flat plate as a polyhedra
O.bodies.append(
```

```
polyhedra_utils.polyhedra(
poly_mat,
v=((-r_plate, -r_plate, -thickness / 2), (r_plate, -r_plate, -thickness / 2),
(r_plate, r_plate, -thickness / 2), (-r_plate, r_plate, -thickness / 2),
(-r_plate, -r_plate, thickness / 2), (r_plate, -r_plate, thickness / 2),
(r_plate, r_plate, thickness / 2), (-r_plate, r_plate, thickness / 2)),
fixed=True,
color=(0.35, 0.35, 0.35),
)
)

# Create a pack of exactly 217 spherical particles on the plate surface
# Create an empty sphere pack
sphere_pack = pack.SpherePack()

# Generate 217 spheres with random radii between 1mm and 10mm
for _ in range(50):
radius = random.uniform(0.001, 0.01)  # Random radius between 1 mm and 10mm
center = (
random.uniform(-r_plate, r_plate),  # X position within the plate
random.uniform(-r_plate, r_plate),  # Y position within the plate
random.uniform(thickness, thickness + 0.05)  # Z position slightly above the plate
)
sphere_pack.add(center, radius)



# Add the spheres to the simulation
for center, radius in sphere_pack:
O.bodies.append(utils.sphere(center, radius, material=poly_mat))

```

```
# Controller to add a sphere after 20 seconds
sphereAdded = False

def controller(generateSphereAt=1):  # Sphere will be generated after one second.
global sphereAdded
if not sphereAdded and O.time > generateSphereAt:  # Add sphere after a certain time
sphereAdded = True
frictMat = FrictMat(young=186E9, density=16678, poisson=0.34)
O.materials.append(frictMat)
sph = utils.sphere((0.0, 0.0, 1.0), .00822, material=frictMat)
O.bodies.append(sph)
sph.state.vel = (0, 0, -250)

escapedParticles = 0  # Counter for particles that escape

# Add the functions to track Vmax and escaped particles
def trackVelocityAndEscaped():
global escapedParticles
maxVelocity = 0  # To find Vmax
escaped = 0  # Temporary counter for escaped particles this iteration

for b in O.bodies:
if isinstance(b.shape, Sphere):  # Only track spheres
vel = b.state.vel.norm()  # Magnitude of velocity
maxVelocity = max(maxVelocity, vel)

# Check if the particle has flown out of bounds
if b.state.pos[2] > 0.05 or b.state.pos[2] < -0.02:  # Adjusted escape condition
escaped += 1

escapedParticles += escaped  # Add to total count
plot.addData(
i=O.iter,
Vmax=maxVelocity,
escapedParticles=escapedParticles
# Changed to track Vmax instead of escaped particles
```

```
)

# Simulation engines
O.engines = [
ForceResetter(),
InsertionSortCollider([Bo1_Polyhedra_Aabb(), Bo1_Facet_Aabb(),
 Bo1_Sphere_Aabb()]),
InteractionLoop(
[Ig2_Facet_Polyhedra_PolyhedraGeom(),
Ig2_Polyhedra_Polyhedra_PolyhedraGeom(),
 Ig2_Sphere_Polyhedra_ScGeom()],
[Ip2_PolyhedraMat_PolyhedraMat_PolyhedraPhys(),
 Ip2_FrictMat_FrictMat_FrictPhys()],
[Law2_PolyhedraGeom_PolyhedraPhys_Volumetric(),
 Law2_ScGeom_FrictPhys_CundallStrack()]
),
NewtonIntegrator(gravity=(0, 0, -g), damping=0.5),
PyRunner(command='controller(generateSphereAt=1)', virtPeriod
 =0.1),
PyRunner(command='trackVelocityAndEscaped()', iterPeriod=100)
]

O.trackEnergy = True

# Time step
O.dt = 0.5 * PWaveTimeStep()

plot.plots = {'i': ('Vmax', 'EscapedParticle')}
plot.plot()

# Save the simulation state
O.saveTmp()
```

LISTING 3: Sphere YADE Simulation Code 1-10 mm

```
from yade import pack, polyhedra_utils, geom, utils, qt, export,
  plot
import math

global sphereAdded, escapedParticles
sphereAdded = False
```

```

# Parameters
g = 0.00011  # Gravity (m/s^2)
d_plate = .1 #Diameter of the flat plate (m)
r_plate = d_plate / 2  # Radius of the flat plate
thickness = 0.01 # Thickness of the plate (m)

# Material for the polyhedral particles
poly_mat = PolyhedraMat(density=1282, young=7.233E9, poisson
  =0.3, frictionAngle=0.6)
plate_mat = FrictMat(young=7.4e9, poisson=0.25, frictionAngle=
  math.radians(30), density=1200)

O.materials.append(poly_mat)
O.materials.append(plate_mat)

# Create the flat plate as a polyhedra
O.bodies.append(
polyhedra_utils.polyhedra(
poly_mat,
v=((-r_plate, -r_plate, -thickness / 2), (r_plate, -r_plate, -
  thickness / 2),
(r_plate, r_plate, -thickness / 2), (-r_plate, r_plate, -
  thickness / 2),
(-r_plate, -r_plate, thickness / 2), (r_plate, -r_plate,
  thickness / 2),
(r_plate, r_plate, thickness / 2), (-r_plate, r_plate, thickness
   / 2)),
fixed=True,
color=(0.35, 0.35, 0.35),
)
)

# Create a pack of exactly 217 spherical particles on the plate
  surface
sphere_pack = pack.SpherePack()
sphere_pack.makeCloud(
(-r_plate , -r_plate , thickness ),  # Bottom aligned with the
  plate surface
(r_plate , r_plate, thickness + 0.015),
```

```
rMean=0.01,  # Mean radius of 1 mm
num=217,  # Specify the exact number of spheres
seed=32
)

# Add the spheres to the simulation
for center, radius in sphere_pack:
O.bodies.append(utils.sphere(center, radius, material=poly_mat))

# Controller to add a sphere after 20 seconds
sphereAdded = False

def controller(generateSphereAt=1):  # Sphere will be generated
 after one second.
global sphereAdded
if not sphereAdded and O.time > generateSphereAt:  # Add sphere
 after a certain time
sphereAdded = True
frictMat = FrictMat(young=186E9, density=16678, poisson=0.34)
O.materials.append(frictMat)
sph = utils.sphere((0.0, 0.0, 1.0), .00822, material=frictMat)
O.bodies.append(sph)
sph.state.vel = (0, 0, -250)

escapedParticles = 0  # Counter for particles that escape

# Add the functions to track Vmax and escaped particles
def trackVelocityAndEscaped():
global escapedParticles
maxVelocity = 0  # To find Vmax
escaped = 0  # Temporary counter for escaped particles this
 iteration

for b in O.bodies:
if isinstance(b.shape, Sphere):  # Only track spheres
vel = b.state.vel.norm()  # Magnitude of velocity
maxVelocity = max(maxVelocity, vel)

# Check if the particle has flown out of bounds
```

```
if b.state.pos[2] > 0.05 or b.state.pos[2] < -0.02:  # Adjusted
 escape condition
escaped += 1

escapedParticles += escaped  # Add to total count
plot.addData(
i=O.iter,
Vmax=maxVelocity,
escapedParticles=escapedParticles
# Changed to track Vmax instead of escaped particles
)

# Simulation engines
O.engines = [
ForceResetter(),
InsertionSortCollider([Bo1_Polyhedra_Aabb(), Bo1_Facet_Aabb(),
 Bo1_Sphere_Aabb()]),
InteractionLoop(
[Ig2_Facet_Polyhedra_PolyhedraGeom(),
Ig2_Polyhedra_Polyhedra_PolyhedraGeom(),
 Ig2_Sphere_Polyhedra_ScGeom()],
[Ip2_PolyhedraMat_PolyhedraMat_PolyhedraPhys(),
 Ip2_FrictMat_FrictMat_FrictPhys()],
[Law2_PolyhedraGeom_PolyhedraPhys_Volumetric(),
 Law2_ScGeom_FrictPhys_CundallStrack()]
),
NewtonIntegrator(gravity=(0, 0, -g), damping=0.5),
PyRunner(command='controller(generateSphereAt=1)', virtPeriod
 =0.1),
PyRunner(command='trackVelocityAndEscaped()', iterPeriod=100)
]

O.trackEnergy = True
O.dt = 0.5 * PWaveTimeStep()

plot.plots = {'i': ('Vmax', 'EscapedParticle')}
plot.plot()
```

```
# Save the simulation state
O.saveTmp()
```

LISTING 4: Sphere YADE Simulation Code 10 mm

Turnitin Report

SIMULATION OF EJECTA DYNAMICS ON RYUGU'S SURFACE INDUCED BY VERTICAL IMPACTS USING YADE

ORIGINALITY REPORT

4%	3%	2%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Heinz Konietzky. "Numerical Modeling in Micromechanics via Particle Methods", CRC Press, 2017 Publication	<1%
2	Submitted to University of Hertfordshire Student Paper	<1%
3	pure.tue.nl Internet Source	<1%
4	science.gsfc.nasa.gov Internet Source	<1%
5	eprints.utm.my Internet Source	<1%
6	physicsinfilm.wordpress.com Internet Source	<1%
7	Submitted to University of Southampton Student Paper	<1%
8	mobt3ath.com Internet Source	<1%

9	15462.courses.cs.cmu.edu Internet Source	<1 %
10	N. A. Artemieva, V. V. Shuvalov. "Numerical simulation of high-velocity impact ejecta following falls of comets and asteroids onto the Moon", Solar System Research, 2008 Publication	<1 %
11	www.kct.ac.in Internet Source	<1 %
12	Submitted to University of Birmingham Student Paper	<1 %
13	Y. Kishino. "Powders and Grains 2001", A.A.Balkema Lisse/Abingdon/Exton (pa)/Tokyo, 2020 Publication	<1 %
14	Van Zeebroeck, M.. "Determining tangential contact force model parameters for viscoelastic materials (apples) using a rheometer", Postharvest Biology and Technology, 200408 Publication	<1 %
15	Vasileios Angelidakis, Katia Boschi, Karol Brzeziński, Robert A. Caulk et al. "YADE - an extensible framework for the interactive simulation of multiscale, multiphase, and multiphysics particulate systems", Computer Physics Communications, 2024	<1 %

Publication

16	Y. Tang, D.S. Lauretta, R.-L. Ballouz, D.N. DellaGiustina, C.A. Bennett, K.J. Walsh, D.R. Golish. "Simulating impact-induced shaking as a triggering mechanism for mass movements on Bennu", Icarus, 2023 Publication	<1 %
17	www.ijisae.org Internet Source	<1 %
18	M. Oda, K. Iwashita. "Mechanics of Granular Materials - An Introduction", A.A. Balkema/Rotterdam/Brookfield, 2020 Publication	<1 %
19	Submitted to Cardiff University Student Paper	<1 %
20	dspace.dtu.ac.in:8080 Internet Source	<1 %
21	www.pace-nsc.org Internet Source	<1 %
22	Martin Haustein, Anton Gladkyy, Rüdiger Schwarze. "Discrete element modeling of deformable particles in YADE", SoftwareX, 2017 Publication	<1 %
23	developer.mulesoft.com Internet Source	<1 %

24	etd.aau.edu.et Internet Source	<1 %
25	inesc-id.pt Internet Source	<1 %
26	personal.ems.psu.edu Internet Source	<1 %
27	Eric M. MacLennan, Joshua P. Emery. "Thermophysical Investigation of Asteroid Surfaces. II. Factors Influencing Grain Size", The Planetary Science Journal, 2022 Publication	<1 %
28	Takaya Okamoto, Kosuke Kurosawa, Hidenori Genda, Takafumi Matsui. "Impact Ejecta near the Impact Point Observed using Ultra-high-speed Imaging and SPH Simulations, and a Comparison of the Two Methods", Journal of Geophysical Research: Planets, 2020 Publication	<1 %
29	airandspacelaw.olemiss.edu Internet Source	<1 %
30	essay.utwente.nl Internet Source	<1 %
31	fenix.tecnico.ulisboa.pt Internet Source	<1 %
32	phaidra.univie.ac.at Internet Source	<1 %

33	uec.repo.nii.ac.jp Internet Source	<1%
34	hdl.handle.net Internet Source	<1%
35	Anand Handa, Rohit Negi, S. Venkatesan, Sandeep K. Shukla. "Implementing Enterprise Cyber Security with Open-Source Software and Standard Architecture", River Publishers, 2023 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

Curriculum Vitae

	Basic Information
Name	Talal Mohammed Ali Bin Snkar
Place of Birth	Jeddah, Kingdom of Saudi Arabia
Date of Birth	20, 03, 2001
Address	No. 57, Section 2, Linsen Rd, Huwei Township, Yunlin County, 632
Year	**Education**
2021 - present	International University Liaison Indonesia (IULI)
2015 - 2018	Al'Thager high school
2024 - Present	National Formosa University
Year	**Courses**
2024	Basic Python
2024	Basic Latex
2024	Toeic
2023	Connected Leadership
Year	**Seminars & Workshops**
2023	Basic Aviation Eng. Practical Training
Year	**Work Experiences**
2017	Sales Man For Nuts Splendor
2023	Copywriting
2023	Content Creation
2023	Marketing Advisor
2024	Comparison of Standards Practices for Aircraft Maintenance
2023	AIAA Design Competition

2024	Crypto Trading