

Prediction of particle mixing in rotary drums by DEM data-driven models

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur
(Dr.-Ing.)

von **M.Sc. Wencong Wu**

geb. am 04.06.1993 in Hubei, China

genehmigt durch die Fakultät für Verfahrens- und Systemtechnik
der Otto-von-Guericke-Universität Magdeburg

Promotionskommission	Prof. Dr.-Ing. habil. Ulrich Krause (Vorsitz)
	Prof. Dr.-Ing. habil. Evangelos Tsotsas (Gutachter)
	Prof. Dr.-Ing. Andreas Bück (Gutachter)
	Prof. Dr.-Ing. Harald Kruggel-Emden (Gutachter)

eingereicht am: 31.03.2025

Promotionskolloquium am: 30.06.2025

Abstract

Particle mixing is a fundamental unit operation in many industrial applications. Rotary drums are widely adopted due to their simple structure, yet particle mixing within them exhibits high complexity due to the interplay of multiple influencing factors. This complexity makes accurate prediction particularly challenging. While discrete element method (DEM) simulations have been extensively validated through experiments and can effectively model this process, their high computational cost remains a significant limitation. Therefore, finding efficient and reliable approaches to predict particle mixing in rotary drums based on a limited number of DEM simulations while covering a multi-factor parameter space is of critical importance.

This study presents two types of predictive approaches: the first, referred to as cross-correlation, and the second, based on machine learning models. Both approaches rely on experimentally validated DEM data. Additionally, the subdomain-based mixing index (SMI) was chosen as the sole quantitative description of mixing processes in this study.

Cross-correlation establishes a relationship between the mixing time of spherical particles in a 2D drum at different revolution frequencies and that in a 3D drum. In other words, as a bridge, it enables the prediction of computationally expensive 3D simulations using cheap 2D simulations. Once developed, cross-correlation demonstrated strong predictive performance for mixing time in a 3D drum at different revolution frequencies, achieving $R^2 = 0.92$. Remarkably, even when predicting mixing time for density ratios, which are beyond the parametric range of the training set, the model maintained acceptable accuracy with $R^2 = 0.86$. The reasons for its limited performance in predicting different size ratios were analyzed. The primary advantages of the cross-correlation method lie in its strong interpretability, expandability, and its ability to deliver satisfactory predictions even with a highly limited dataset.

Regarding machine learning approaches, the particle swarm optimized support vector regression (PSO-SVR) model was first employed to predict the mixing of spherical particles in rotary drums. Compared to cross-correlation, this model incorporated two additional influencing factors—size ratio and drum length—alongside of the revolution frequency and density ratio. Furthermore, it was capable of predicting not only the mixing time but also the mixing degree at steady mixing state. The predictive accuracy reached $R^2 = 0.95$ for mixing time and $R^2 = 0.90$ for mixing degree. While running a new DEM

simulation requires several hours, the PSO-SVR model, including hyperparameter tuning, could be trained and make predictions within just tens of seconds, significantly reducing computational time.

Building upon this, machine learning was further extended to predict the mixing of non-spherical particles in rotary drums. Three different models—artificial neural network (ANN), extremely randomized trees (ERT), and PSO-SVR—were employed to analyze the mixing of rod-like particles, which were represented using the multi-sphere method. Their simulation results were compared with those of spherical particles. Each machine learning model exhibited distinct strengths and weaknesses. A comprehensive comparison across various dimensions, such as predictive accuracy, interpretability, and implementation complexity, was presented in a table. Overall, the predictive performance of these models was satisfactory, and their total modeling time was significantly shorter than that required for running new DEM simulations.

Kurzzusammenfassung

Teilchenmischung ist eine grundlegende Einheitoperation in vielen industriellen Anwendungen. Rotierende Trommeln werden aufgrund ihrer einfachen Struktur weit verbreitet eingesetzt, doch die Teilchenmischung in ihnen zeigt eine hohe Komplexität aufgrund des Zusammenspiels mehrerer Einflussfaktoren. Diese Komplexität macht eine genaue Vorhersage besonders herausfordernd. Während Simulationen mit der diskreten Elementemethode (DEM) durch Experimente umfassend validiert wurden und diesen Prozess effektiv modellieren können, bleibt ihr hoher Rechenaufwand eine bedeutende Einschränkung. Daher ist es von entscheidender Bedeutung, effiziente und zuverlässige Ansätze zu finden, um die Teilchenmischung in rotierenden Trommeln basierend auf einer begrenzten Anzahl von DEM-Simulationen vorherzusagen und dabei einen multifaktoriellen Parameterraum abzudecken.

Diese Studie stellt zwei Arten von Vorhersageansätzen vor: den ersten, als Kreuzkorrelation bezeichnet, und den zweiten, der auf maschinellen Lernmodellen basiert. Beide Ansätze beruhen auf experimentell validierten DEM-Daten. Zudem wurde der subdomänenbasierte Mischungsindex (SMI) als einzige quantitative Beschreibung der Mischungsprozesse in dieser Studie gewählt.

Die Kreuzkorrelation stellt eine Beziehung zwischen der Mischzeit von sphärischen Partikeln in einer 2D-Trommel bei verschiedenen Umdrehungsfrequenzen und der in einer 3D-Trommel her. Mit anderen Worten: Als Brücke ermöglicht sie die Vorhersage von rechnerisch aufwendigen 3D-Simulationen mithilfe kostengünstiger 2D-Simulationen. Nach ihrer Entwicklung zeigte die Kreuzkorrelation eine starke Vorhersageleistung für die Mischzeit in einer 3D-Trommel bei verschiedenen Umdrehungsfrequenzen mit $R^2 = 0.92$. Bemerkenswerterweise behielt das Modell selbst bei der Vorhersage von Mischzeiten für Dichteverhältnisse, die außerhalb des Parameterbereiches des Lernsatzes lagen, eine akzeptable Genauigkeit mit $R^2 = 0.86$ bei. Die Gründe für die begrenzte Leistung bei der Vorhersage unterschiedlicher Größenverhältnisse wurden analysiert. Die Hauptvorteile der Kreuzkorrelationsmethode liegen in ihrer starken Interpretierbarkeit, Erweiterbarkeit und der Fähigkeit, auch mit einem stark begrenzten Datensatz zufriedenstellende Vorhersagen zu liefern.

Im Hinblick auf maschinelle Lernansätze wurde zunächst das Partikelschwarm-optimierte Support Vector Regression (PSO-SVR) Modell zur Vorhersage der Mischung von sphärischen

Partikeln in rotierenden Trommeln eingesetzt. Im Vergleich zur Kreuzkorrelation integrierte dieses Modell zwei zusätzliche Einflussfaktoren—Größenverhältnis und Trommellänge—neben Umdrehungsfrequenz und Dichteverhältnis. Darüber hinaus war es in der Lage, nicht nur die Mischzeit, sondern auch den Mischungsgrad im stationären Mischzustand vorherzusagen. Die Vorhersagegenauigkeit erreichte $R^2 = 0.95$ für die Mischzeit und $R^2 = 0.90$ für den Mischungsgrad. Während die Durchführung einer neuen DEM-Simulation mehrere Stunden erfordert, konnte das PSO-SVR-Modell einschließlich Hyperparameter-Tuning innerhalb von nur wenigen Sekunden trainiert werden und Vorhersagen treffen, wodurch die Rechenzeit erheblich reduziert wurde.

Darauf aufbauend wurde die Anwendung künstlicher Intelligenz weiter ausgedehnt, um die Mischung von nicht-sphärischen Partikeln in rotierenden Trommeln vorherzusagen. Drei verschiedene Modelle—künstliches neuronales Netzwerk (ANN), extrem zufällige Bäume (ERT) und PSO-SVR—wurden eingesetzt, um die Mischung von stabförmigen Partikeln zu analysieren, die mit der Multi-Sphären-Methode dargestellt wurden. Ihre Simulationsergebnisse wurden mit denen sphärischer Partikel verglichen. Jedes maschinelle Lernmodell wies unterschiedliche Stärken und Schwächen auf. Ein umfassender Vergleich hinsichtlich verschiedener Dimensionen wie Vorhersagegenauigkeit, Interpretierbarkeit und Implementierungskomplexität wurde in einer Tabelle dargestellt. Insgesamt war die Vorhersageleistung dieser Modelle zufriedenstellend, und ihre gesamte Modellierungszeit war erheblich kürzer als die für die Durchführung neuer DEM-Simulationen erforderliche Zeit.

Contents

1	Introduction	1
1.1	Overview	1
1.2	Solid particle mixing	1
1.3	Particle mixing in rotary drums	7
1.3.1	Forms of transverse bed motion	7
1.3.2	Radial segregation in rotary drums	12
1.3.3	Axial segregation in rotary drums	15
1.4	Discrete element method (DEM) simulation	16
1.4.1	DEM models for spherical particles	17
1.4.1.1	Equations of particle motion	17
1.4.1.2	Contact model	18
1.4.1.3	Numerical integration algorithm	24
1.4.2	DEM models for non-spherical particles	26
1.4.2.1	Representation of particle orientation	26
1.4.2.2	Representation of particle shape	29
1.4.3	Determination of DEM parameters	32
1.4.3.1	Simulation parameters	32
1.4.3.2	Physical properties	36
1.5	Mixing quantification	37
1.5.1	Mixing indices	37
1.5.1.1	Lacey index	38
1.5.1.2	Entropy of mixing	39
1.5.1.3	Subdomain-based mixing index (SMI)	40
1.6	Motivation and objectives	41
2	Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation	43
3	Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model	92

4	Prediction of rod-like particle mixing in rotary drums by three machine learning methods based on DEM simulation data	132
5	Summary and Outlook	147
5.1	Summary	147
5.2	Outlook	149

Normenclature

Symbols

a, b, c	Shape parameter	-
\mathbf{a}	Unit axis of rotation	-
A	Area	m^2
\mathbf{A}	Rotation matrix	-
e	Coefficient of restitution	-
E	Young's modulus	Pa
El	Mixing entropy index	-
f	Filling degree	-
f_k	Participation factor in SMI	-
\mathbf{F}	Force	N
Fr	Froude number	-
\mathbf{g}	Gravitational acceleration	m/s^2
G	Shear modulus	Pa
H	Bed length	m
I, \mathbf{I}	Moment of inertia	$\text{kg} \cdot \text{m}^2$
k	Elastic constant	kg/s^2
LI	Lacey index	-
m	Mass	kg
n	Revolution frequency	rpm
n_1, n_2	Blockiness parameter	-
\mathbf{n}	Unit normal vector	-
N	Number	-
p	Number fraction of particles	-
P	Modified fraction in SMI	-
q	Quaternion	-
Q	Types of particle	-
r	Radius of particle	m
r_{neigh}	Neighbor search radius	m
r_f	Radius of particle	m

r_s	Skin distance	m
R	Radius of rotary drum	m
R^2	Coefficient of determination	-
s_{bin}	Bin size	m
S	Entropy of mixing	-
SMI	Subdomain mixing index	-
t	Time	s
Δt	Time step	s
T	Torque	N · s
v	Translational velocity of particle	m/s
x	Center position of particle	-
X, Y, Z	Coordinates in body-fixed system	-

Greek symbols

α	Damping coefficient	-
β	Rotation angle	rad
γ	Damping constant	kg/s
δ	Overlap of two particles	m
δ	Displacement of two particles	m
ϵ	Filling angle	rad
θ	Dynamic angle of repose	rad
μ	Coefficient of friction	-
ν	Poisson's ratio	-
ρ	Density	kg/m ³
σ^2	Variance	-
ϕ	Angle of rolling resistance	rad
ω, ω	Angular velocity of particle	rad/s

Subscripts

c	Contact
$crit$	Critical
eq	Equivalent
i, j	Indices
m	Mixed
max	Maximum
n	Normal
p	Particle
r	Relative
$roll$	Rolling

s	Selected region
t	Tangential
tot	Total
u	Unmixed
W	Wall
x, y, z	Axes of space-fixed coordinate system
X, Y, Z	Axes of body-fixed coordinate system

Superscripts

b	Body-fixed coordinate system
-----	------------------------------

Abbreviations

ANN	Artificial neural network
CFD	Computational fluid dynamics
DEM	Discrete element method
ERT	Extremely randomized trees
GPU	Graphics processing unit
MPT	Magnetic particle tracking
MRI	Magnetic resonance imaging
PEPT	Positron emission particle tracking
PSO	Particle swarm optimization
PSO-SVR	Particle swarm optimized support vector regression
RPT	Radioactive particle tracking
SVR	Support vector regression
SMI	Subdomain-based mixing index

Chapter 1

Introduction

1.1 Overview

This chapter provides an informative introduction to solid particle mixing in rotary drums. It first describes the fundamentals of solid mixing and segregation phenomena, with a particular focus on their underlying mechanisms. Next, the various forms of transverse bed motion and the unique segregation behaviors observed in rotary drums are introduced. Subsequently, the most commonly used numerical simulation method for granular flow—the discrete element method (DEM)—is explained in detail. This is followed by a discussion on the quantitative description of the mixing process using mixing indices. Finally, the motivation and objective of this dissertation are outlined.

1.2 Solid particle mixing

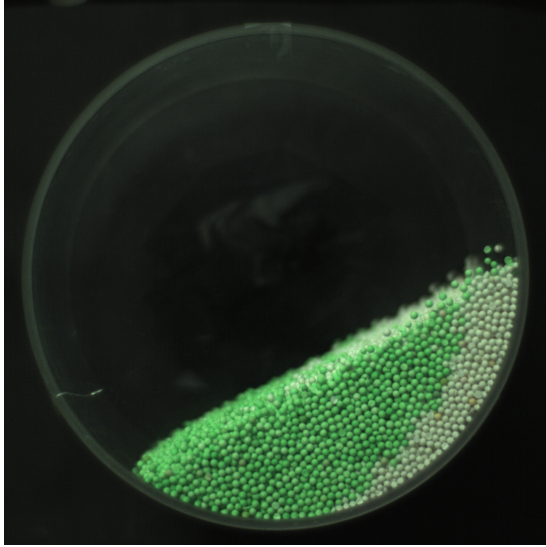
Solid particle mixing is an ancient and widely applied technical operation, it has been performed for thousands of years, long before modern science and industrial processes emerged. For example, in early metallurgy, mixing metal ores with charcoal was an important part of smelting and metalworking processes; mixing clay, sand, and other natural materials was central to pottery making and building materials in ancient times; ancient cultures like the Chinese and Greeks mixed minerals and plant extracts to create cosmetics, dyes, and pigments. Solid particle mixing is also quite common in our daily life. A typical example is baking, where ingredients like flour, sugar, salt, and baking powder need to be evenly mixed; otherwise, the cake or bread may rise inconsistently, or parts of the product may be too sweet while others lack flavor. More importantly, solid particle mixing is one of the most fundamental unit operations in many industries, including chemical manufacturing, food processing, material engineering and pharmaceuticals. The uniformity of mixing is crucial in industrial processes, as it often impacts the quality of the final product. This is especially evident in pharmaceutical industry, where insufficient

active ingredient content can render the drug ineffective, while excessive amounts may be fatal.

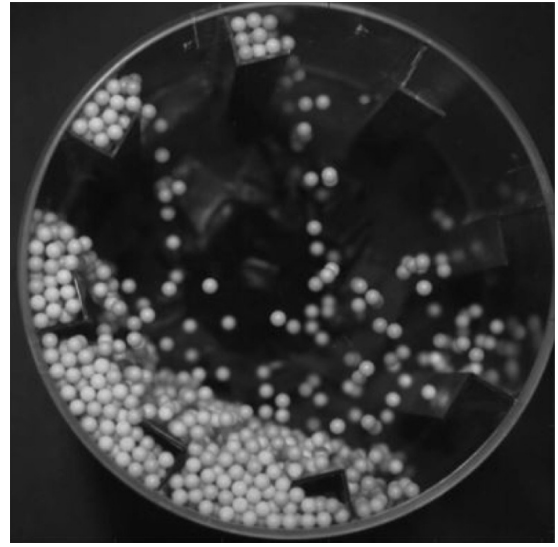
Despite the commonality and importance of solid particle mixing, research in this field has remained limited to this day, and understanding the dynamic behavior of particles during the mixing process remains a significant challenge in physics, with many unresolved problems. However, due to the seemingly apparent nature of solid particle mixing, it is often mistakenly believed that mixing of solids is much simpler and certainly more intensively studied than that of fluids, while in reality, the opposite is true. In fact, solid particle mixing is significantly different in principle from fluid mixing. Under normal circumstances, liquids and gases could mix spontaneously due to molecular thermal motion, known as Brownian motion. This kind of motion was found by Robert Brown in 1827. Molecules in these phases move randomly and collide with each other, allowing substances to diffuse throughout the system and achieve uniform mixing. However, for solid particles, to be more specific, for collections of discrete macroscopic solid materials with sizes from micrometer to centimeter range, the Brownian motion is irrelevant. Since the internal thermal fluctuations are insufficient to move solid particles, external forces such as gravity or mechanical action from mixing paddles are necessary. It was not until 1954 that Lacey [1] first proposed three possible mechanisms for solid particle mixing, which have now been widely recognized by the academic community. These three principal mechanisms are:

- 1) Diffusive mixing: the distribution of particles over a freshly developed surface.
- 2) Convective mixing: the transfer of groups of adjacent particles from one location in the bulk to another.
- 3) Shear mixing: the setting up of slipping planes within the bulk.

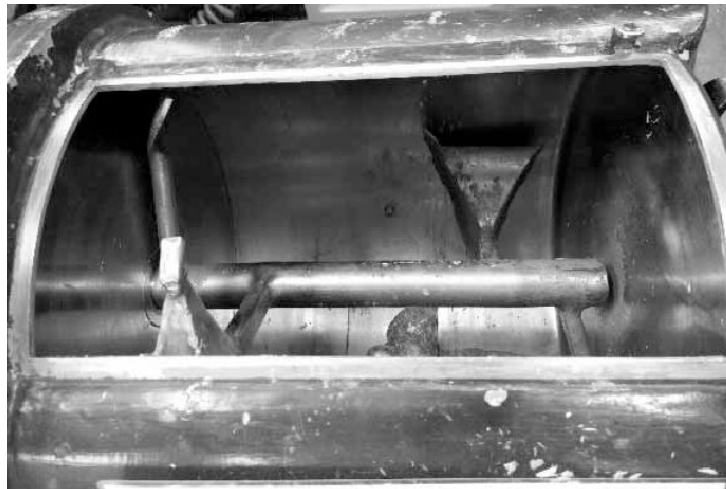
Diffusive mixing always exists in any type of mixer as long as the inside particles are kept in a state of free flow. This mechanism serves as the primary mechanism in some simple mixers, such as horizontal rotary drums. When the inside particle bed has a stable dynamic angle of repose, diffusive mixing keeps taking place in the inclined plane, as shown in Fig. 1.1(a). For convective mixing, the movement of solids is forced by mixing tools. Simple modifications to the basic rotary drum, for example, adding L-shaped flights to its inner wall, can make convective mixing dominant. The flights continuously separate, lift, and release portions of the particle bed, accelerating the overall mixing process within the flighted rotary drum, as illustrated in Fig. 1.1(b). In shear mixing, a high-speed mixing tool is employed to induce the slipping of particle groups past one another. When a mixture contains some undesirable agglomerates that should be broken down to achieve a good mix, shear forces are particularly effective. Ploughshare mixers are a typical mixing device that primarily utilizes the shear mixing mechanism. Figure 1.1(c) gives the picture of a ploughshare mixer.



(a)



(b)



(c)

Figure 1.1: Mixers based on different solid mixing mechanisms: (a) rotary drum—diffusive mixing; (b) flighted rotary drum [2]—convective mixing; (c) ploughshare mixer [3]—shear mixing.

Another common misconception about solid particle mixing is that for non-cohesive particles, the type of mixer only affects the speed of mixing, and given enough time, the particles will eventually reach a uniform mix regardless of the mixer used. In reality, achieving a uniformly mixed state for solid particles during production is not a simple task that can be accomplished by merely extending the mixing time. This challenge is closely related to a unique phenomenon of solid materials: segregation. Although liquids or gases can sometimes undergo stratification due to density differences, such as the classic separation of oil and water, segregation in solids is not only caused by more complex factors but also manifests itself in a variety of forms. Specifically, when solid particles have differences in material properties, especially in size, density or shape, their mixture tends to segregate rather than mix uniformly [4, 5]. Besides material properties, device and operation parameters, handling processes, and environmental conditions also influence segregation [6].

Segregation phenomena are quite common in processes involving the handling of solid particulate materials. When a mixture containing a range of particle sizes is poured to form a heap, the finer fraction deposits preferentially near the center and the coarser fraction near the edges. Sometimes stratification may also occur, with interleaved layers of coarser and finer particles. Figure 1.2(a) illustrates an example of segregation and stratification occurring during the formation of a quasi-2D bounded heap. When filling a storage bin under gravity, the smaller particles concentrate at the center of the bin while the larger particles concentrate at the sides of the bin, as shown in Fig. 1.2(b). The most famous example of solid segregation must be the Brazil nut effect. When observing a serving of mixed nuts, one will often find the larger Brazil nuts appearing on the upper surface, as indicated in Fig. 1.2(c). Even if this is not the case at first, by gently shaking the can for a short period of time, the larger Brazil nuts will magically move to the top of the entire nut mixture.

However, the most complex and diverse segregation patterns are observed in rotary drums. In the radial direction, smaller or denser particles tend to concentrate in the center of the particle bed; in some special cases, radial streaks may occur. In the axial direction, alternating particle bands will form in a long drum because of the particle size difference. Those patterns can be found in Fig. 1.3.

Mosby et al. [12] reviewed ten main segregation mechanisms, short descriptions and alternative terms are given below:

- 1) Rolling: rounder or larger particles are more likely to roll down along the surface of a heap.
- 2) Sieving: smaller particles tend to move downward through a rolling or sliding layer of larger particles. This mechanism is often accompanied by percolation, which can be considered a distinct segregation mechanism in its own right.

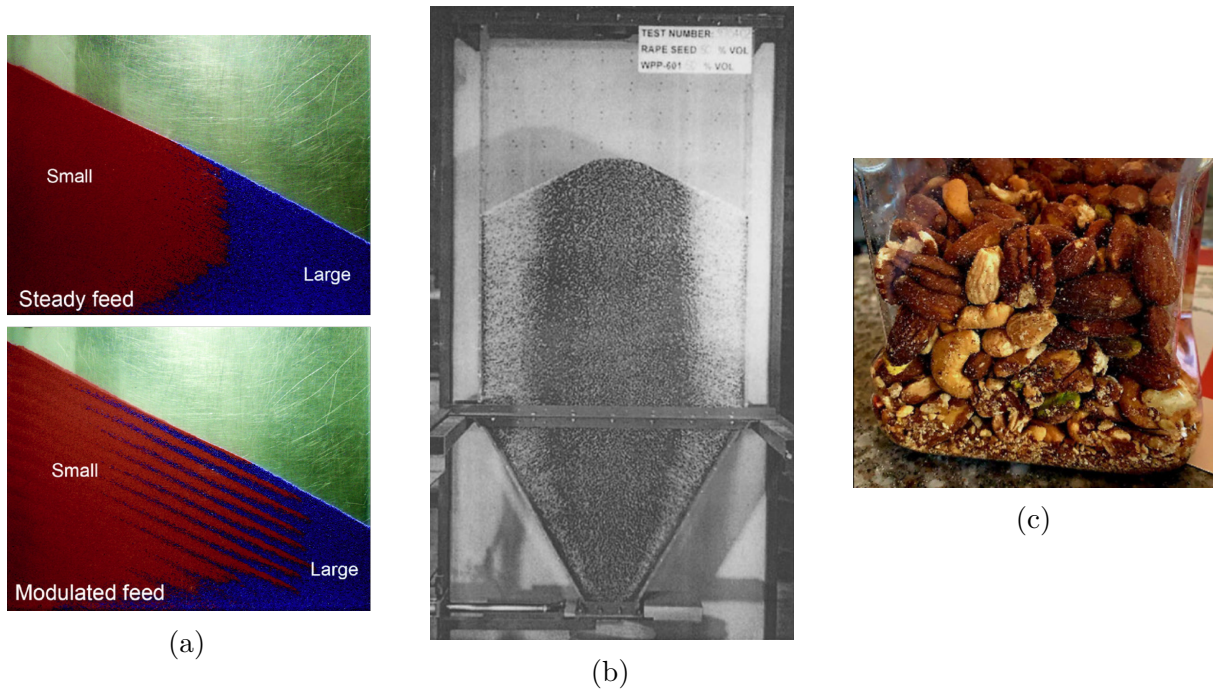


Figure 1.2: Different forms of solid particle segregation: (a) segregation and stratification when particles build up a quasi-2D bounded heap [7]; (b) segregation during gravity filling of a storage bin [8]; (c) Brazil nut effect [9].

- 3) Percolation: this mechanism causes smaller particles to fall through gaps between larger particles due to localized shear. While similar to sieving, it does not require the larger particles to be in a flowing layer. The mechanism can be triggered by vibrations or local shear planes and may also result in the upward displacement of larger particles.
- 4) Push-away effect: when heavier particles are positioned above lighter, gravity causes the heavier particles to push the lighter particles aside. This results in heaps where heavier particles settle near the center of the heap and lighter ones near the edges.
- 5) Angle of repose effects: this mechanism involves differences in the angles of repose of various components filled sequentially onto a heap. The component with a lower angle of repose flows over that with a higher angle, moving toward the edges of the heap.
- 6) Displacement: larger particles rise to the surface of a mixture of large and small particles in stages, as a result of vibrations with vertical amplitudes. Small particles percolate into the vacated space and prevent large ones from returning. The mechanism has also been called “floating migration”.
- 7) Trajectory effects: the retarding force in a fluid decelerates smaller particles in free motion more significantly than larger (or heavier) particles. This mechanism can lead to segregation when particles move through the air, e.g. when they fall from the end of a conveyor belt. It is more accurately referred to as “inertia segregation”.

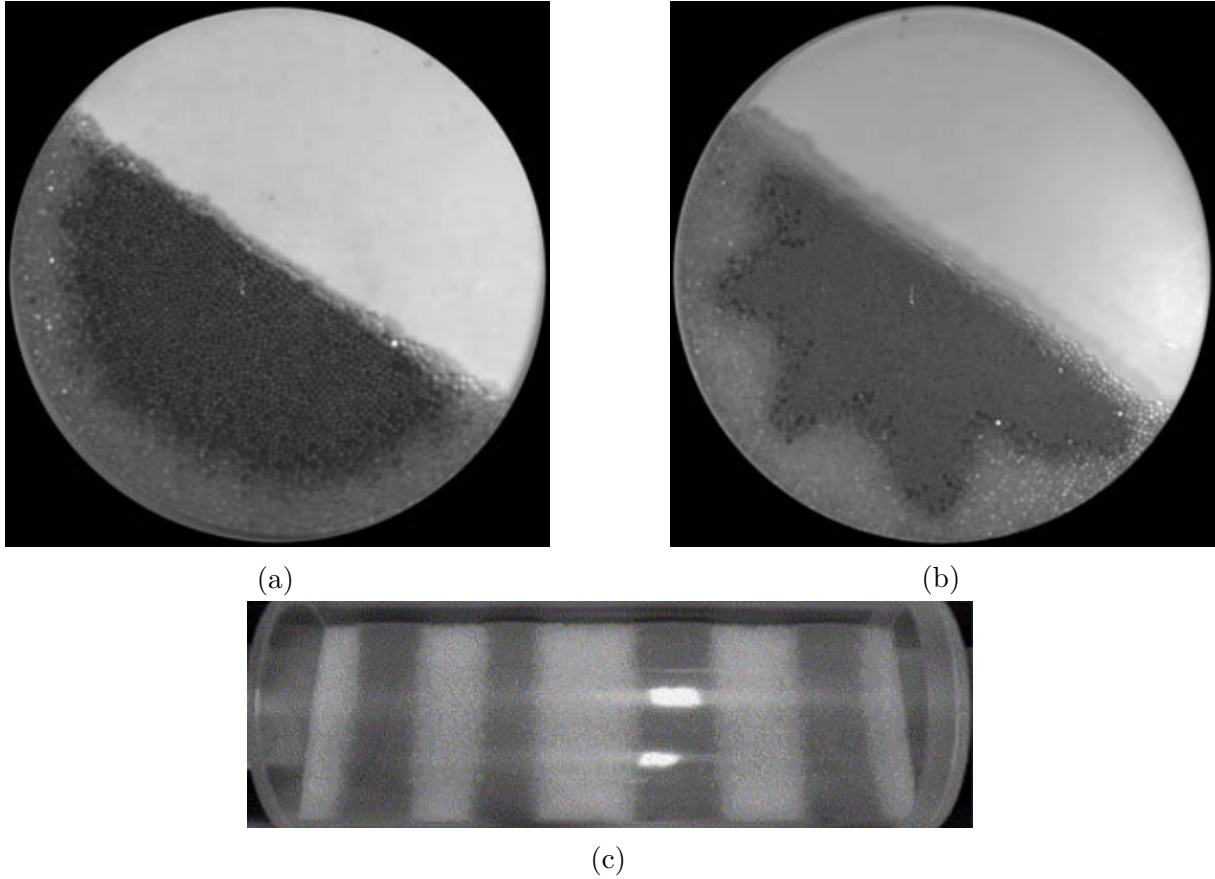


Figure 1.3: Different patterns of particle segregation in rotary drums: (a) smaller and denser particles form a core [10]; (b) smaller and denser particles form radial streaks [10]; (c) alternating particle bands in a long rotary drum [11].

- 8) Air current effects: this mechanism is induced by specific handling conditions, for example during the filling of a silo. As particles fall, they generate an air current circulating from the center toward the walls, which carries very fine particles ($< \text{ca. } 50 \mu\text{m}$) away from the center and deposits them along the silo walls.
- 9) Fluidization effects: the fluidizable component of lighter or finer particles may become fluidized when falling, and then float on the heap surface.
- 10) Impact effects: occurs when particles with higher coefficients of restitution bounce off the surface of a heap, causing them to be distributed further from the center compared to particles with lower coefficients. Increasing the filling rate will significantly enhance this mechanism. Could also be called “bouncing”.

From the above summary of segregation mechanisms, it can be observed that segregation phenomena in particle handling processes are highly complex, often involving multiple mechanisms acting simultaneously, and sometimes in competition with each other. However, the ten segregation mechanisms mentioned above seem inadequate in fully explaining the particle segregation phenomena within rotary drums. This will be discussed in detail in

Section 1.3.

Additionally, different segregation mechanisms are significantly influenced by particle properties and operational parameters. In the heap segregation shown in Fig. 1.2(a), at least four mechanisms—rolling, sieving, angle of repose effects, and impact effects—work together. With the help of the discrete element method (DEM) simulation, Zhang et al. [13] obtained a comprehensive view of the particle movements during heap segregation, and found that some special areas with extremely low porosity and high resistance to flow also play an important role during the formation of segregation or stratification. They also investigated the effects of size ratio, injection height, and mass ratio on the segregation. Regarding the segregation in the bin shown in Fig. 1.2(b), sieving is always considered as the primary segregation mechanism. However, through DEM simulation, Zhang et al. [14] reproduced the original experiment quite accurately and found that bouncing of small particles and rolling of large particles are main segregation mechanisms. The effect of friction coefficients was also examined. These two examples demonstrate the significant role of DEM simulation in particle research. This simulation method not only provides deeper insights into segregation by tracking the motion of individual particles, but it is also highly effective in scenarios where particle properties and operational parameters frequently change, offering considerable flexibility for analysis. A detailed introduction into DEM simulations will be presented in Section 1.4.

1.3 Particle mixing in rotary drums

As the simplest mixer in terms of structure, rotary drums are widely popular in a variety of industries involving the mixing of solid particles. Although various drum types exist, this study focuses on cylindrical drums with their rotational axes aligned horizontally, perpendicular to gravity, and without internal flights. In this section, the transverse motion of a particle bed in rotary drums is first introduced. Then the segregation phenomena in radial and axial direction as well as their possible mechanisms are reviewed separately.







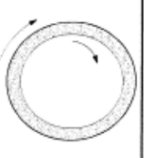
1.3.1 Forms of transverse bed motion

As a rotary drum turns around its axis, the mixing of particles occurs in both the transverse plane and the axial direction. The axial mixing is considered to be a random motion as the results of interparticle collisions and can be described by Fick's law [15]. This kind of diffusive mixing in the axial direction is much slower than the particle mixing in the transverse plane, and its effect is not obvious in relatively short drums, so most of the research works were concentrated on the transverse motion.

In a stationary drum, the inside particles form a bed of solids at the bottom. While the

drum rotates, the particle bed takes on different forms. Based on an experimental study, Henein et al. [16] summarized six transverse bed motion regimes: slipping, slumping, rolling, cascading, cataracting and centrifuging. They also pointed out that these regimes depend on many variables such as revolution frequency, filling degree, drum diameter, particle size, and particle shape. Mellmann [17] studied the transition between different flow regimes in depth, and developed mathematical models to predict these transitions. He further subdivided the slipping regime into sliding and surging, and clarified the effect of Froude number, filling degree and wall friction coefficient on the transverse motion in rotary drums, as summarized in Table 1.1.

Table 1.1: Regimes of transverse motion of solid particles in rotary drums [17].

Basic form	Slipping motion		Cascading ("tumbling") motion			Cataracting motion	
Subtype	Sliding	Surging	Slumping	Rolling	Cascading	Cataracting	Centrifuging
Schematic							
Physical process	Slipping		Mixing			Crushing	Centrifuging
Froude number Fr [-]	$0 < Fr < 10^{-4}$		$10^{-5} < Fr < 10^{-3}$	$10^{-4} < Fr < 10^{-2}$	$10^{-3} < Fr < 10^{-1}$	$0.1 < Fr < 1$	$Fr \geq 1$
Filling degree f [-]	$f < 0.1$	$f > 0.1$	$f < 0.1$	$f > 0.1$		$f > 0.2$	
Wall friction coeff. μ_w [-]	$\mu_w < \mu_{w,c}$	$\mu_w \leq \mu_{w,c}$	$\mu_w > \mu_{w,c}$			$\mu_w > \mu_{w,c}$	
Application	no use		Rotary kilns and reactors; rotary dryers and coolers; mixing drums			Ball mills	no use

The dimensionless quantity in Table 1.1—the Froude number Fr —is considered to be the characteristic criterion for the transverse motion of solid bed in rotary drums, which is defined as the ratio of centrifugal force to gravity:

$$Fr = \frac{\omega^2 R}{g}, \quad (1.1)$$

where R is the radius of the rotary drum, g is the gravitational acceleration and ω is the angular velocity of the rotary drum, with the unit of rad/s. The conversion between the angular velocity ω and the more commonly used revolution frequency n (with the unit of rpm) can be expressed as:

$$\omega = \frac{2\pi n}{60}. \quad (1.2)$$

The filling degree f is defined as the portion of the drum occupied by the particle bed.

Note that the particle bed includes not only the solids but also the voids between particles. Assuming a flat bed surface in a stationary drum, based on the geometric relationships shown in Fig. 1.4, the filling degree can be converted from the volume ratio to the area ratio, and directly correlated to the bed height H :

$$f = \frac{A_{\text{solid bed}}}{A_{\text{drum}}} = \frac{\varepsilon R^2 - R \sin \varepsilon R \cos \varepsilon}{\pi R^2} = \frac{\varepsilon - \sin \varepsilon \cos \varepsilon}{\pi}, \quad (1.3)$$

with

$$\varepsilon = \arccos\left(\frac{R - H}{R}\right). \quad (1.4)$$

The filling angle ε here corresponds to the half bed angle of the circular segment occupied by the solid bed, with the unit of rad. Moreover, the parameter $\mu_{W,c}$ in Table 1.1, denoted here by $\mu_{W,crit}$, represents the critical wall friction coefficient for the transition from slipping motion to non-slipping motion, which will be described in detail after the introduction of different regimes.

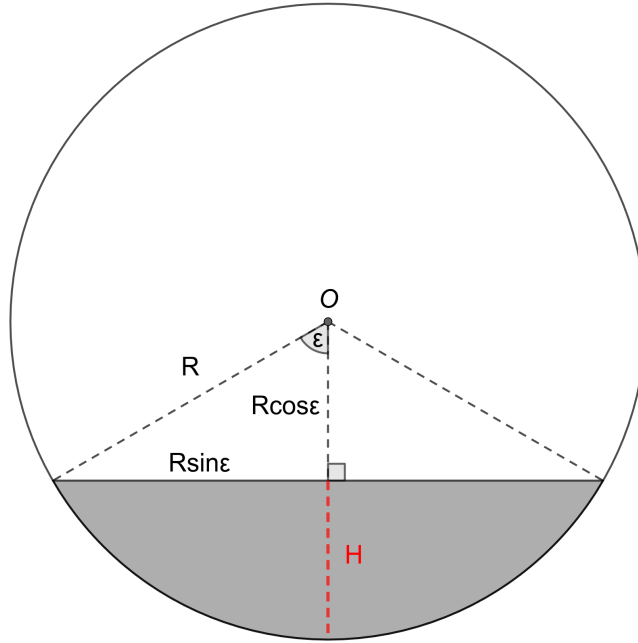


Figure 1.4: Schematic of the cross-section of the solid particle bed in a stationary drum.

Two subtypes of slipping—sliding and surging—both occur in slow rotated drums with smooth inner walls. When the filling degree is really low, sliding may be observed, which is characterized by the particle bed constantly sliding down the wall with a very small inclination. Increasing the filling degree, sliding turns into surging. The surging regime is characterized by the whole bed briefly adhering to the wall, being lifted, and then sliding back, repeatedly cycling between these two states. Slipping is an undesirable motion as nearly no particle mixing takes place. Specifically, the bed moves as a whole, and its

inclination is too small to form a continuous particle flow on its surface. Therefore, this motion state should be avoided in practice by increasing the roughness of the inner wall, such as by adhering sandpaper, to enhance mixing quality.

As opposed to slipping, centrifuging is another extreme motion that occurs at very high revolution frequencies. This motion regime is characterized by the fact that the entire solid bed remains fixed to the drum wall as a uniform film. When centrifuging begins to appear, the centrifugal and gravitational forces acting on the particles are in equilibrium, i.e. $Fr = 1$. Substituting $Fr = 1$ into Eq. 1.1 and combining with Eq. 1.2, the expression of the critical revolution frequency n_{crit} can be determined as:

$$n_{crit} = \frac{30}{\pi} \sqrt{\frac{g}{R}}. \quad (1.5)$$

This critical parameter can be used to specify the revolution frequency of rapidly rotating drums such as ball mills. Hence, the Froude number Fr can be expressed in a more convenient form:

$$Fr = \left(\frac{n}{n_{crit}} \right)^2. \quad (1.6)$$

Additionally, the critical revolution frequency can also be taken as a reference for other motion regimes.

Slumping, rolling, cascading, and cataracting (subtypes in Table 1.1) are four non-slipping motion regimes that occur when the revolution frequency ranges from small to the critical value. The cataracting regime is commonly observed in ball mills for grinding particles, where particles follow parabolic trajectories after detaching from the particle bed. The other three regimes are the main forms of motion in rotary drums used for mixing, drying or cooling.

Slumping occurs in the low-filled drum ($f < 0.1$) at low revolution frequencies (less than 3% of the critical revolution frequency). It is characterized by the continuous elevation of the particle bed by the rotating drum, with particles on the sloped surface periodically avalanching from the upper half to the lower half. Slumping clearly differs from surging since the particle bed no longer moves as a whole, and particle flow is generated during surface avalanching.

Both rolling and cascading may take place in the range of 1% to 30% of the critical revolution frequency. As can be seen in Table 1.1, there is an overlap in the Fr intervals corresponding to these two flow regimes. The rolling regime is characterized by a thin layer of continuously fast-moving particles on a sloped flat upper bed surface, while the majority of the bed is rotating with the wall at the revolution frequency at the bottom. As the revolution frequency increases further, the steady flow observed in the rolling regime becomes progressively unstable, with the flat bed surface deforming into an S-shaped profile. This new flow regime is referred to as cascading. The transition from rolling to cascading is also closely related to the length of the rotary drum. Yang et al. [18] found in

their DEM simulations that higher revolution frequencies are needed for longer drums to achieve this flow regime transition. Their findings can be explained by the influence of end walls. Taberlet et al. [19] demonstrated that the friction on end walls can greatly affect the particle flow properties, and this effect is more pronounced in short drums compared to long ones.

Rolling is considered as the ideal motion regime for particle mixing since it provides a stable and efficient particle flow. Most studies on particle mixing in rotary drums have focused on the rolling regime. Similarly, the majority of cases examined in this study were conducted within this regime. Only a few instances of mixing in short drums operating at high revolution frequencies exhibited cascading behavior. The constant slope of the bed surface in the rolling regime is defined as the dynamic angle of repose θ . The critical wall friction coefficient $\mu_{W,c}$ in Table 1.1 is also related to this angle. Mellmann [17] derived an expression for the critical wall friction coefficient based on the momentum balance around the axis of rotation as:

$$\mu_{W,crit} = \frac{2 \sin^3 \varepsilon \sin \theta}{3\pi f(1 + Fr)}. \quad (1.7)$$

Notably, the critical wall friction coefficient here represents the friction coefficient between the particles and the cylindrical wall, as the effects of the two end walls were not considered in his calculations.

Based on the granular flow characteristics in the rolling regime, the particle bed can be divided into two distinct regions: an active layer near the free surface, which flows rapidly downward under gravity, and a passive layer deeper in the bed, rotating slowly at the same revolution frequency as the drum [20, 21, 22]. These two layers are separated by a curved boundary line, as illustrated in Fig. 1.5. Particles at the boundary line AWB have zero velocity parallel to the inclined surface. According to experimental observations and simulations [23, 24], the particle velocity profiles along the mid-chord of the particle bed exhibit bi-linear distributions. However, the turning point of the two linear velocity distributions is not located on the boundary line but slightly below it, a detail that is also shown in Fig. 1.5. It is generally accepted that particles do not mix in the passive layer. Mixing takes place when particles cross the upper boundary line (WA) into the active layer. After undergoing mixing in the active layer, particles are transported back into the passive layer through the lower boundary line (WB). The center of the symmetric boundary line is referred to as the vortex point W . Particles at the vortex point have zero velocity in all directions.

Another noteworthy aspect is the filling degree of rotary drums. When drums are used for mixing in practice, their filling degrees are relatively low, typically between 20% to 50%. This is because too high filling degree hinders particle mixing in rotary drums. Xiao et al. [25] observed in DEM simulations that, as the filling degree increases from 20% to 45%, the thickness of the active layer (CW in Fig. 1.5) decreases. However, since the

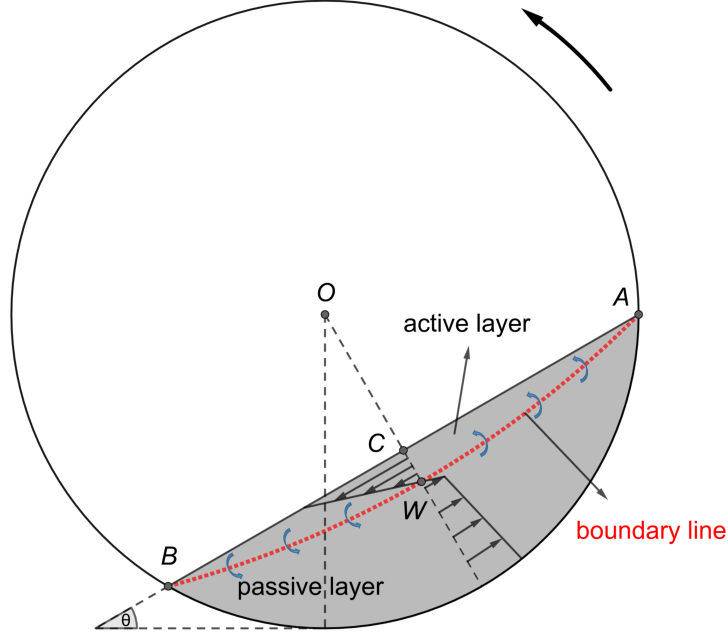


Figure 1.5: Schematic of the solid particle bed in the rolling regime.

chord length of the particle bed (ACB in Fig. 1.5) also increases, the reduction in the total area of the active layer in the cross-section is less pronounced. Nonetheless, the overall particle bed expands significantly with higher filling degrees, leading to a noticeable decrease in the proportion of the active layer relatively to the whole particle bed. This ultimately results in longer mixing times. When the filling degree exceeds 50%, the active layer lies above the axis of rotation, and a dead zone can be observed around the axis [26]. The particles in the dead zone always behave as an unmixed compact core, even if the revolution frequency is increased (within the rolling regime). Hlosta et al. [27] performed DEM simulations of rotary drums with filling degrees ranging from 10% to 80% and found that either too low or too high a filling degree led to inhomogeneous mixing.

Understanding the unique particle motion in the rolling regime forms the foundation for investigating segregation phenomena in rotary drums, which will be explored in the following subsections.

1.3.2 Radial segregation in rotary drums

When there are differences in particle properties, such as size, density, or shape, the mixture tends to segregate rather than mix, and this phenomenon is also observed in rotary drums. Segregation occurs in both radial and axial directions of rotary drums, whereas typically radial segregation is usually observed in quasi-2D drums, i.e., when the axial length of drums is rather short, as this minimizes the effect of axial segregation.

In the rolling regime, as the drum rotates, smaller or heavier particles gather at the

center of the drum, surrounded by larger or lighter particles, as shown in Fig. 1.3(a). Whereas size segregation can be attributed to the sieving mechanism [28, 29], density segregation can be explained by the buoyancy effect [30, 31]. The sieving mechanism has already been described in Section 1.2, that is, in a flowing layer, the small particles tend to be sieved downward into the voids of the larger particles, causing the larger particles to rise to the free surface. The buoyancy effect is actually a density-driven push-away effect, and also operates within a flowing layer. The heavier particles sink to lower levels in the layer, whereas lighter particles rise, resulting in segregation across the layer. Obviously, the “buoyancy” mechanism here is named by analogy to fluid dynamics.

However, to fully explain the core-shell structure of segregation formed in rotary drums, it is necessary to combine these segregation mechanisms with the unique particle motion in the rolling regime. The active layer in the rolling regime is a typical flowing layer where both sieving and buoyancy take effect. Smaller or heavier particles drift downward in the active layer, displacing larger or lighter particles upward. Due to the continuous rotation of the drum, larger or lighter particles floating on the surface of the active layer move rapidly until they enter the passive layer at the lower boundary line, away from the vortex point. These particles then travel along the periphery of the particle bed within the passive layer, and eventually re-enter the active layer from the upper boundary line, also distant from the vortex point. In contrast, smaller or heavier particles located deeper in the active layer tend to approach the vortex point. Once near the vortex point, their velocities decrease and their circulation paths become shorter, leading to their entrapment in the vicinity of the vortex point. These processes are amplified each time particles pass through the active layer, until a stable core-shell structure is formed. The radial segregation caused by size or density differences occurs quite quickly, often in a few revolutions of the drum.

As for particles with different shapes, their segregation is caused by the differences in particle flowability, essentially the rate of energy dissipation during flowing [32]. Particles with poor flowability will concentrate in the center of the particle bed to form a core, while particles with good flowability will form a shell on the periphery of the bed. Compared to size or density segregation, shape segregation is much weaker, as it takes longer to reach a steady state and results in a fuzzier interface between segregated domains. In addition, the shape segregation occurs along the surface of the active layer rather than through the active layer. Take the mixing of spherical particles and cubical particles as an example. Due to the poor flowability, only a few cubical particles are able to roll down along the inclined surface of the active layer all the way to the bottom end. Most of the cubical particles come to rest near the middle of the inclined surface and are overtaken by rapidly moving spherical particles. This forces the cubical particles to sink into the passive layer and finally form a core.

Another interesting observation is that, when comparing size and density segregation under identical conditions for particles of different shapes, spherical particles always exhibit

the highest segregation extent among any other non-spherical particles [33, 34, 35]. This observation can also be explained by their flowability differences. The lower flowability of non-spherical particles impedes the sieving and buoyancy mechanisms within the active layer, thereby weakening the overall segregation effect. Moreover, Kumar et al. [34, 35] found in both experiments and DEM simulations that, for non-spherical particles with identical particle density but different shapes, higher packing densities result in weaker size segregation. This is because higher packing density corresponds to lower interparticle porosity, which also inhibits the sieving mechanism.

The complexity of radial segregation in rotary drums lies in the fact that different mechanisms can be effective simultaneously and lead to seemingly contradictory results. For example, He et al. [33] found in DEM simulations that when there is a huge difference in shape between spherical and ellipsoidal particles, reverse segregation may occur, i.e., ellipsoids in the periphery and spheres in the center. They explained that when ellipsoidal particles are too flat or too elongated, their packing becomes much looser than that of spheres. When rolling down in the active layer, the spheres are more likely to be sieved through the voids instead of rolling over the stacks of ellipsoids. Another example of reverse segregation is that when the size ratio is above five, large particles may segregate in the center. Thomas [36] explained that when the size ratio is very large, the push-away effect driven by mass cannot be ignored, and the large particles will sink into the small particles when flowing in the active layer. The combination of size segregation and density segregation is of particular interest and practical relevance. Jain et al. [10] mixed large-heavy steel beads with small-light glass beads in a rotary drum, causing the sieving and buoyancy mechanisms to oppose each other, successfully eliminating segregation.

Radial streak patterns in rotary drums (Fig. 1.3(b)) is another complex and interesting phenomenon that remains poorly understood. The conditions for the formation of streaks are quite demanding. Radial streaks can only be observed in nearly half-filled (filling degree = 50%) quasi-2D drums. The revolution frequency of the drum should be really low, approaching the lower limit of maintaining the rolling regime, typically 1 rpm. Radial streak patterns can be induced solely by size difference [37] or density difference [38], but the required ratios are relatively high, with a lower limit of approximately 3 in both cases. However, if size segregation and density segregation enhance rather than oppose each other, the ratios required to produce radial streaks will decrease [10]. Moreover, radial streak pattern has been successfully reproduced by DEM simulations [39, 40]. Liao et al. [38] and Pereira et al. [39] both believe that the difference in the dynamic angle of repose between the two types of particles also plays a key role in the formation of radial streaks.

1.3.3 Axial segregation in rotary drums

Due to the much slower particle flow in the axial direction compared to that in the transverse plane, axial segregation takes significantly longer to develop than radial segregation [28]. Nevertheless, axial segregation has a considerable impact on the particle distribution in long drums. Once axial segregation forms, radial segregation becomes less apparent. Differences in particle density and size both lead to axial segregation, but their manifestations are noticeably different.

When particles with varying densities are mixed in a long drum, lighter particles tend to completely envelop the heavier ones in all directions. As a result, the core of heavy particles becomes almost invisible, even when the cross-section is observed from the transparent ends of the drum. However, if a cross-section at the midpoint of the axis is examined (which can be easily achieved in DEM simulations), the internal particle bed reveals a core-shell structure consistent with that observed in quasi-2D drums [41].

Axial segregation caused by differences in particle size is characterized by the formation of alternating bands of large and small particles alongside of the drum, as shown in Fig.1.3(c). All experiments on axial segregation indicate that large particles always accumulate near the end walls of rotary drums [42, 43, 44]. When the aspect ratio of the drum, defined as the ratio of its length to its diameter, is close to or slightly less than 1, three relatively stable bands with an alternating large-small-large particle arrangement are typically formed [41, 45]. In contrast, in longer drums with an aspect ratio greater than 2, not only does the number of particle bands increase with drum length, but the width of these bands also becomes unstable and slowly changes over time [11, 43]. The mechanism behind the formation of particle bands remains unclear. Early studies often attributed it to differences in the dynamic angle of repose between large and small particles [42, 46]. However, this explanation fails to answer why particles with different densities do not lead to the formation of alternating particle bands. In more recent studies, particularly those utilizing DEM simulations, researchers have gradually recognized the indispensable role of end walls in the formation of axial particle bands. The particle flow near the end walls was found to be significantly different from the flow far from end walls [47, 48]. Cui et al. [49] created a rotary drum with frictionless end walls through DEM simulation and observed a phenomenon never seen in experiments before, where small particles accumulated near the end walls. This confirmed that the friction on the end walls is the cause of large particle accumulation. Arntz et al. [45] set the rotary drum to be non-closed periodically continued in the DEM simulation and found that the axial particle bands completely disappeared, particles of different sizes exhibiting only radial segregation. This further illustrates that the end walls play a crucial role in the axial distribution of particles throughout the drum.

1.4 Discrete element method (DEM) simulation

Discrete element method (DEM) simulations have been mentioned quite a few times in previous sections. Since the DEM was developed by Cundall and Strack [50], it has become the most popular and effective simulation method for studying granular flow. As its name suggests, the DEM is a typical discrete model. By modeling each single particle as a distinct entity, DEM tracks individual motions of particles over time by calculating forces and torques arising from particle collisions and external fields. The overall particulate system behavior results from individual particle interactions. Using this modeling strategy, DEM can provide detailed insights into particle dynamics, such as the trajectories or instantaneous velocities of individual particles, which is extremely hard, or even impossible to obtain from physical experiment.

Another major advantage of DEM lies in its ability to combine with other models or simulation methods. For example, the DEM can incorporate particle-particle contact heat transfer models to predict the temperature variations in stirred beds [51, 52]. The DEM can also be linked with pore network models to investigate the influence of liquid phase distributions on damage and deformation of particle aggregates during convective drying [53]. The coupling of DEM and computational fluid dynamics (CFD) has become a well-established simulation approach for studying multi-phase flows, such as modeling the particle residence time distribution [54] and coating uniformity [55] in fluidized beds. This flexibility expands the scope of DEM far beyond granular flow alone.

Despite its strengths, DEM has certain limitations. Since DEM tracks the motion and interactions of individual particles, the computational cost grows significantly, typically between linear and quadratic, as the number of particles increases. For each time step, DEM must compute the contact forces between all particle pairs within the interaction range, as well as their resulting accelerations, velocities, and displacements. This makes simulations of industrial-scale systems highly time-consuming and resource-intensive. Furthermore, the accuracy of DEM simulations often requires very small time steps to ensure numerical stability, particularly when dealing with stiff particle materials or high-speed dynamics. A typical time step for DEM is between 10^{-4} and 10^{-6} s, which can be as much as 100 times smaller than the time steps commonly used in CFD. As a result, even short-duration processes require millions of iterations, further compounding the computational demand. To mitigate these challenges, researchers have developed techniques such as coarse graining [56, 57], optimized contact detection algorithms [58, 59, 60], parallel computing [61], and graphics processing unit (GPU) acceleration [62, 63]. While these advancements significantly reduce computation times, they may introduce trade-offs such as reduced

accuracy, increased implementation complexity, or specialized hardware requirements, potentially limiting their widespread adoption. Moreover, with current computational power, simulating typical industrial-scale systems involving billions of particles using DEM remains far beyond reach. Therefore, minimizing the time consumption associated with DEM while obtaining accurate information about particulate systems remains an urgent need in current research.

In this section, the DEM models for spherical particles are first presented, which includes governing equations, contact model and numerical integration algorithm. Then, two popular models for representing non-spherical particles are introduced, and the differences between modeling spherical and non-spherical particles are also discussed. Finally, the detailed process of determining DEM parameters is explained. This process is critical in DEM simulations, as it directly influences the accuracy and reliability of the results.

1.4.1 DEM models for spherical particles

The open-source software LIGGGHTS [64] was employed for all DEM simulations in this study. This software is a fork of the classical molecular dynamics simulator LAMMPS [65]. In DEM, a wide variety of sub-models are available to represent particle interactions and system behavior. The equations provided in this section are those specifically applied in this study.

Due to the perfect symmetry of a sphere in all directions, DEM modeling for spherical particles is relatively straightforward. Many sub-models developed for spherical particles can be directly applied to non-spherical particles or require only minor modifications for adaptation. Notably, the original DEM model developed by Cundall and Strack [50] was designed for 2D discs, whereas the widely adopted DEM models today focus on 3D spheres. Therefore, this section provides a detailed introduction to the modeling of spherical particles as a foundation.

1.4.1.1 Equations of particle motion

Each particle in a granular flow can have two types of motion: translational and rotational. In DEM, the translational and rotational accelerations of a particle are governed by Newton's second law, which are then numerically integrated over a time step to update the particle's translational and angular velocity as well as its position. For a particle i with mass m_i and moment of inertia I_i , its translational and rotational motion can be described as follows:

$$m_i \frac{d^2 \mathbf{x}_i}{dt^2} = m_i \frac{d\mathbf{v}_i}{dt} = \sum \mathbf{F}_{i,c} + m_i \mathbf{g}, \quad (1.8)$$

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum \mathbf{T}_{i,c}, \quad (1.9)$$

where \mathbf{x}_i , \mathbf{v}_i and $\boldsymbol{\omega}_i$ are the center position, translational velocity and angular velocity of particle i , respectively. The translational motion of particle i is a function of the sum of contact forces $\sum \mathbf{F}_{i,c}$ acting on the particle i by other particles or walls as well as the non-contact gravitational force $m_i \mathbf{g}$. Although other types of contact or non-contact forces, such as the particle-fluid interaction force in multi-phase flow or capillary force describing the action between wet particles, can also be integrated into the governing equations of DEM, they are not considered in this study. Similar to the translational motion, the rotational motion of particle i depends on the sum of contact torques $\sum \mathbf{T}_{i,c}$ acting on the particle i . The moment of inertia of a solid sphere can be written as:

$$I = \frac{2mr^2}{5}, \quad (1.10)$$

where r is the radius of the solid sphere.

1.4.1.2 Contact model

In DEM, contact models play a crucial role in simulating particle interactions, with the hard-sphere and soft-sphere models being two primary approaches. The hard-sphere model assumes instantaneous interactions between rigid particles without any deformation and can only deal with binary collisions. This model can be classified as an event-driven approach, since it increases the simulation time by processing the collisions one by one according to the contact order [66]. As a result, the hard-sphere model is generally suitable for dilute particle flows, where its assumptions are good approximations and the model remains computationally efficient [67]. However, in dense particle flows, multiple and long-lasting particle contacts are quite common, making the hard-sphere model inadequate. In contrast, the soft-sphere model allows for particle overlap to represent the deformation of particles during contact, even though the particles are still assumed to be rigid. The overlaps are used to calculate elastic and frictional forces for each particle-particle and particle-wall contact, enabling the simulation of multiple and long-lasting particle interactions. As the soft-sphere model always proceeds via small time steps, it is thus referred to as a time-driven approach. In fact, the original DEM framework proposed by Cundall and Strack [50] employs the soft-sphere model as the contact model. Given that the particles in a rotary drum exhibit typical dense particle flow behavior, the DEM simulations in this study also adopt the soft-sphere model.

Many physical parameters in the soft-sphere model are directional. To accurately describe these parameters, the normal direction in the binary spherical particle system is first defined as the unit vector pointing from the center of particle j to the center of particle i :

$$\mathbf{n} = \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|}. \quad (1.11)$$

Two particles are considered to be in contact once the distance between their centers becomes less than the sum of their radii, as shown in Fig. 1.6. The overlap of two spherical particles in the normal direction can be expressed as:

$$\delta_n = \max(0, r_i + r_j - \|\mathbf{x}_i - \mathbf{x}_j\|). \quad (1.12)$$

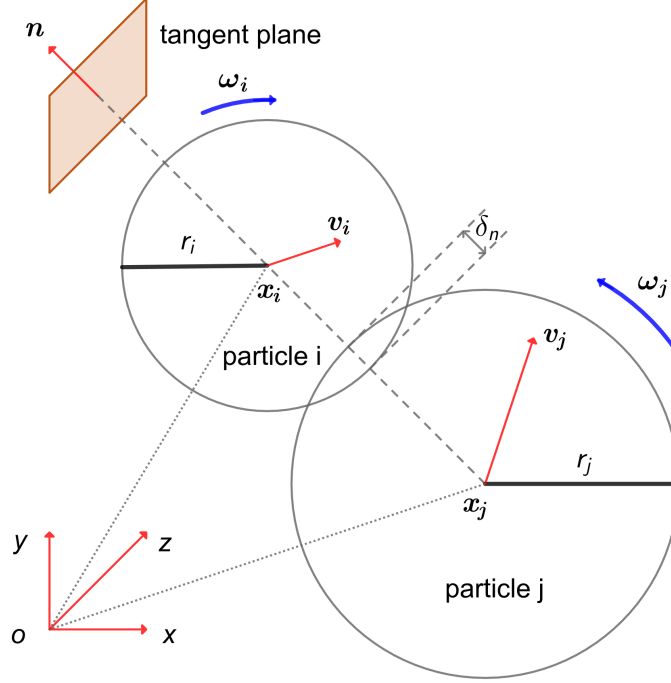


Figure 1.6: Soft-sphere particle-particle contact model with notations.

However, when calculating the relative velocity between two contacting particles, their overlap is ignored, and a single contact point is assumed. The relative velocity at the contact point is determined by considering both the translational velocities and the angular velocities. The angular velocities are converted into linear velocities at the contact point and combined with the translational velocities. Therefore, the relative velocity is calculated as:

$$\mathbf{v}_r = (\mathbf{v}_i - \mathbf{v}_j) + [\boldsymbol{\omega}_i \times (-r_i \mathbf{n}) - \boldsymbol{\omega}_j \times (r_j \mathbf{n})]. \quad (1.13)$$

The relative normal velocity is the projection of the relative velocity vector \mathbf{v}_r onto the unit normal vector \mathbf{n} :

$$\mathbf{v}_{r,n} = (\mathbf{v}_r \cdot \mathbf{n}) \mathbf{n}. \quad (1.14)$$

So the relative tangential velocity is given by:

$$\mathbf{v}_{r,t} = \mathbf{v}_r - \mathbf{v}_{r,n}. \quad (1.15)$$

Since particle i is the subject of the following force analysis, the subscript i is omitted

to simplify the notations. Similar to the velocity decomposition, the contact force from particle j acting on the particles i during collision can also be decomposed into normal and tangential directions:

$$\mathbf{F}_c = \mathbf{F}_n + \mathbf{F}_t. \quad (1.16)$$

All real-world collisions are inelastic, i.e., there is always a loss of kinetic energy during the collision. This energy dissipation can be characterized in DEM using the spring-dashpot model. The model simulates both the elastic and dissipative behaviors of particle collisions by combining a spring and a dashpot in parallel. However, the implementation differs slightly between the normal and tangential directions, as illustrated in Fig. 1.7.

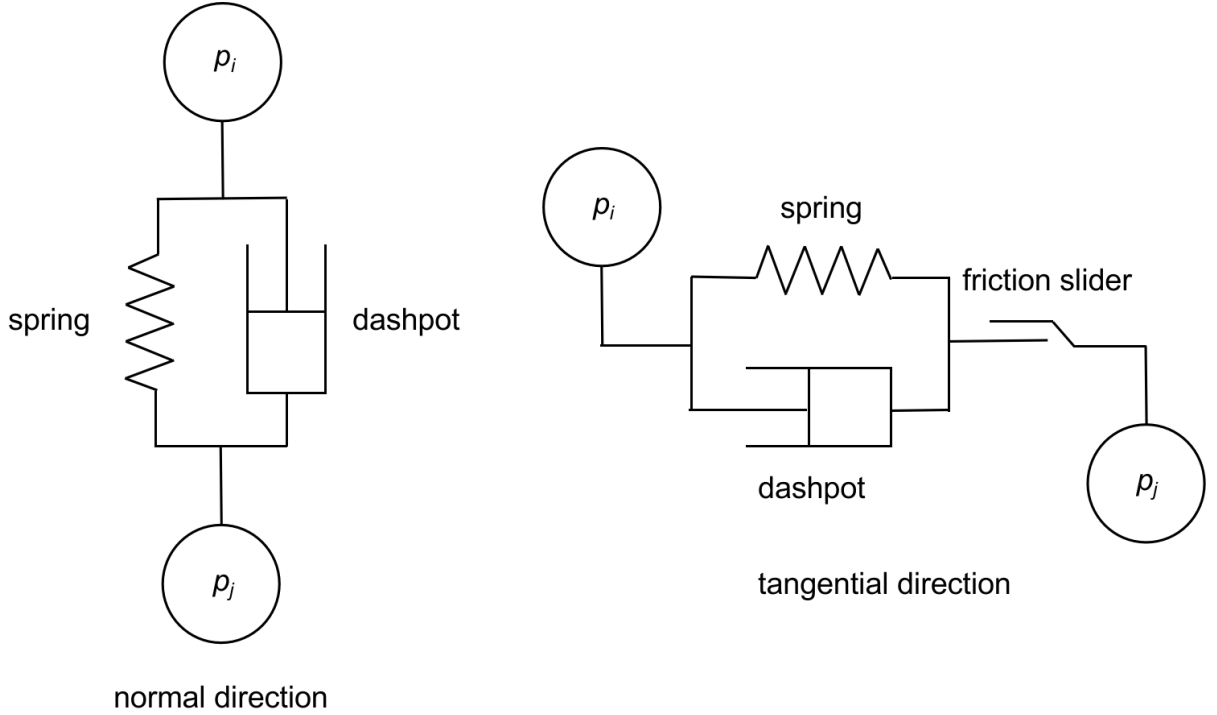


Figure 1.7: Schematic of spring-dashpot model in normal and tangential directions.

Contact force in normal direction

The normal contact force consists of two components: elastic force and damping force, which together account for the deformation and energy dissipation during contact:

$$\mathbf{F}_n = \mathbf{F}_{n,\text{elastic}} + \mathbf{F}_{n,\text{damping}} = k_n \delta_n \mathbf{n} - \gamma_n \mathbf{v}_{r,n}, \quad (1.17)$$

where k_n is the normal elastic coefficient and γ_n is the normal damping coefficient. The normal elastic force term is positive because the direction of the normal elastic force is always from particle j to particle i , which is consistent with the previously defined normal direction \mathbf{n} . The normal damping force term, however, is negative since the normal damping force is always opposite to the normal relative velocity $\mathbf{v}_{r,n}$ to impede the relative

motion of particles.

The simplest spring-dashpot model is the linear spring-dashpot (LSD) model, which was proposed by Cundall and Strack [50]. In the linear model, the elastic force follows Hooke's law, being linearly proportional to the deformation. So the elastic coefficient, also referred to as the stiffness, is a constant value. Similarly, the damping force is assumed to be linearly related to the relative velocity, with the damping coefficient also set as a constant. Due to its simplicity, the linear spring-dashpot model ensures high computational efficiency in the resource-intensive DEM simulations and remains widely popular to this day. However, its physical foundation is not robust, and determining the constants in the model—the elastic coefficient and the damping coefficient—can be challenging. These values are often obtained through experiments, empirical formulas or equivalence between the linear and the nonlinear models [68].

In this study, a more complex and theoretically sound model, known as the Hertzian spring-dashpot (HSD) model, was selected. This is a nonlinear model. In this model, the normal elastic force component is based on Hertzian contact theory. Hertz considered the relationship between the normal elastic force and the normal displacement to be nonlinear, which means that the normal elastic coefficient k_n is also related to the normal overlap δ_n . Unlike the LSD model, where determining the elastic coefficient is challenging, the estimation of elastic coefficient in the HSD model based on material properties has become well-established [69]. The normal elastic coefficient can be written as:

$$k_n = \frac{4}{3}E_{eq}\sqrt{r_{eq}\delta_n}, \quad (1.18)$$

where E_{eq} and r_{eq} are the equivalent Young's modulus and equivalent radius in the binary spherical particle system. The definitions of all the equivalent properties in the HSD model will be given in summary later on.

The normal damping coefficient in this model is not a constant either. Its expression is based on the works by Tsuji et al. [70], who gave a generalized form of the damping coefficient. In the normal direction, the damping coefficient is expressed as:

$$\gamma_n = \alpha(e)\sqrt{k_n m_{eq}}, \quad (1.19)$$

where m_{eq} is the equivalent particle mass, and $\alpha(e)$ is a dimensionless empirical parameter related to the coefficient of restitution e . The coefficient of restitution is defined as the ratio of relative velocity of particles before and after collision. Strictly speaking, this coefficient is not a material property since it also depends on the impact velocity. However, this effect is significant only at high impact velocities (over 10 m/s), so the coefficient of restitution is often taken as a constant in DEM simulations [71]. The analytical solution

of $\alpha(e)$ in HSD model is given by Antypov and Elliott [72]:

$$\alpha(e) = \frac{-\sqrt{5} \ln e}{\sqrt{\ln^2 e + \pi^2}} \quad (0 < e < 1). \quad (1.20)$$

Contact force in tangential direction

In the tangential direction, the spring-dashpot model incorporates both elastic and damping components, similar to those in the normal direction. However, it also accounts for the history of tangential displacement and the Coulomb friction limit. The tangential contact force is written as:

$$\mathbf{F}_t = \mathbf{F}_{t,\text{elastic}} + \mathbf{F}_{t,\text{damping}} = -k_t \boldsymbol{\delta}_t - \gamma_t \mathbf{v}_{r,t}, \quad (1.21)$$

where k_t is the tangential elastic coefficient and γ_t is the tangential damping coefficient. The tangential displacement $\boldsymbol{\delta}_t$, unlike the normal overlap δ_n , not only has a direction but also exhibits a history-dependent effect. It is calculated by integrating the relative tangential velocity at the contact point throughout the duration of contact between two particles:

$$\boldsymbol{\delta}_t = \int_0^t \mathbf{v}_{r,t} dt. \quad (1.22)$$

The direction of the tangential displacement represents the cumulative displacement of the contact point in the current tangent plane relative to the initial contact point over the entire contact history. The elastic force term in Eq. 1.21 is negative because the tangential elastic force is always opposite to the direction of the tangential displacement. The tangential elastic force acts as a restoring force, resisting any further increase in the tangential displacement. The Coulomb criterion limits the maximum value of the tangential contact force, which can be expressed as:

$$\|\mathbf{F}_{t,\text{max}}\| = \mu \|\mathbf{F}_n\|, \quad (1.23)$$

where μ is the coefficient of friction between two particles. Specifically, the tangential contact force between two particles grows according to Eq. 1.21 until it reaches the upper limit. At this point, the particles are considered to be sliding, the tangential displacement is truncated and no longer accumulates. The force is then held at the maximum value until the particles lose contact. Therefore, by combining Eq. 1.23 and Eq. 1.21, the expression of the tangential contact force can be rewritten as:

$$\mathbf{F}_t = \min(-k_t \boldsymbol{\delta}_t - \gamma_t \mathbf{v}_{r,t}, -\mu \|\mathbf{F}_n\| \frac{\mathbf{v}_{r,t}}{\|\mathbf{v}_{r,t}\|}). \quad (1.24)$$

The relationship between the tangential elastic force and the tangential displacement is

also nonlinear in the HSD model. The tangential elastic coefficient is expressed based on the no-slip elastic solution proposed by Mindlin [73]:

$$k_n = 8G_{eq}\sqrt{r_{eq}\delta_n}, \quad (1.25)$$

where G_{eq} is the equivalent shear modulus. Similar to the normal case, the tangential elastic coefficient also depends on the normal overlap δ_n . The expression of tangential damping coefficient follows the generalized form in Eq. 1.19, with a slight adjustment in the dimensionless empirical parameter:

$$\gamma_t = \sqrt{\frac{2}{3}}\alpha(e)\sqrt{k_t m_{eq}}. \quad (1.26)$$

The four equivalent properties (radius r_{eq} , mass m_{eq} , Young's modulus E_{eq} and shear modulus G_{eq}) in the binary spherical particle system with particle i and particle j are summarized as follows [74]:

$$\frac{1}{r_{eq}} = \frac{1}{r_i} + \frac{1}{r_j}, \quad (1.27)$$

$$\frac{1}{m_{eq}} = \frac{1}{m_i} + \frac{1}{m_j}, \quad (1.28)$$

$$\frac{1}{E_{eq}} = \frac{1 - \nu_i^2}{E_i} + \frac{1 - \nu_j^2}{E_j}, \quad (1.29)$$

$$\frac{1}{G_{eq}} = \frac{2(1 + \nu_i)(2 - \nu_i)}{E_i} + \frac{2(1 + \nu_j)(2 - \nu_j)}{E_j}, \quad (1.30)$$

where ν_i and ν_j are the Poisson's ratios of particle i and particle j , respectively. By utilizing equivalent properties, the collision between a particle and a planar wall can be treated similarly by assuming that the wall has an infinite radius and mass. In other words, the equivalent radius and equivalent mass in this case are directly equal to the radius and mass of the particle. It is fairly easy to determine the distance between a spherical particle and a plane in DEM. As long as this distance is smaller than the radius of the spherical particle, the particle is considered to be in contact with the wall.

Contact torque

Following the contact force analysis, the contact torque decomposition is considered. The contact torque has two sources, one from the tangential contact force and the other from rolling friction:

$$\mathbf{T}_c = \mathbf{T}_t + \mathbf{T}_{roll}. \quad (1.31)$$

By definition, the torque generated by the tangential contact force is given by:

$$\mathbf{T}_t = -r_i \mathbf{n} \times \mathbf{F}_t. \quad (1.32)$$

For the torque generated by the rolling friction, the constant directional torque model is applied, which can be expressed as:

$$\mathbf{T}_{roll} = -\mu_{roll}k_n\delta_n\frac{\boldsymbol{\omega}_{r,t}}{\|\boldsymbol{\omega}_{r,t}\|}r_{eq}, \quad (1.33)$$

where μ_{roll} is the rolling resistance coefficient, and $\boldsymbol{\omega}_{r,t}$ is the relative tangential angular velocity. The negative sign in Eq. 1.33 indicates that the torque generated by the rolling friction is always opposite to the relative tangential angular velocity, thereby inhibiting the relative rolling of two particles.

The rolling resistance coefficient is defined by the angle of rolling resistance ϕ :

$$\mu_{roll} = \tan \phi. \quad (1.34)$$

This angle represents the maximum inclination of the slope at which a spherical particle can stay at rest without initiating rolling motion.

The relative angular velocity is defined as follows:

$$\boldsymbol{\omega}_r = \boldsymbol{\omega}_i - \boldsymbol{\omega}_j. \quad (1.35)$$

Since the relative angular velocity is not necessarily in the tangent plane, a projection is needed:

$$\boldsymbol{\omega}_{r,t} = \boldsymbol{\omega}_r - (\boldsymbol{\omega}_r \cdot \mathbf{n})\mathbf{n}. \quad (1.36)$$

Although the constant directional torque model is the simplest among the various models describing rolling resistance, it is quite reliable for characterizing rolling energy dissipation in dynamic flow scenarios, such as particle mixing in rotary drums [75].

1.4.1.3 Numerical integration algorithm

Many equations of motion in DEM are expressed in differential or integral form, such as Eqs. 1.8, 1.9 and 1.22. However, the actual simulation of particle motion in DEM is carried out using explicit numerical integration methods. Explicit integration refers to a numerical approach where the state of the system at the next time step is calculated directly from known quantities at the current step, without solving differential equations iteratively. Among these, the simplest approach is the Euler integration method, which updates particle positions and velocities using linear approximations based on their derivatives at each time step. While straightforward to implement, the Euler method suffers from poor accuracy and numerical instability, making it less suitable for long-term simulations of dynamic systems [76, 77].

In contrast, the velocity Verlet integration algorithm [78, 79] is one of the most widely

used due to its superior accuracy and stability. This algorithm is also the default integration method for spherical particles in LIGGGHTS. By updating velocities and positions in a staggered manner with half-step and full-step updates, the velocity Verlet integration algorithm ensures better energy conservation and reduces numerical errors without significantly increasing computational cost [76].

Particularly, for a single spherical particle, the main procedures of the velocity Verlet integration algorithm are as follows:

- 1) Update the translational and angular velocity (half-step):

The translational velocity of the particle at moment $t + \frac{\Delta t}{2}$ is expressed as:

$$\mathbf{v}(t + \frac{\Delta t}{2}) = \mathbf{v}(t) + \frac{\mathbf{F}(t)}{m} \frac{\Delta t}{2}, \quad (1.37)$$

where Δt represents a full time step used in the DEM simulation, $\mathbf{F}(t)$ is the combined external force, including all the contact forces as well as the gravitational force, on the particle at moment t .

The updating of angular velocity is performed in the same manner:

$$\boldsymbol{\omega}(t + \frac{\Delta t}{2}) = \boldsymbol{\omega}(t) + \frac{\mathbf{T}(t)}{I} \frac{\Delta t}{2}, \quad (1.38)$$

where $\mathbf{T}(t)$ is the combined external torque.

- 2) Update the position (full-step):

The center position of the particle at moment $t + \Delta t$ is updated based on the translation velocity at moment $t + \frac{\Delta t}{2}$:

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \mathbf{v}(t + \frac{\Delta t}{2}) \Delta t. \quad (1.39)$$

- 3) Update force and torque (full-step):

After completing the position update, calculate the force and torque acting on the particle at moment $t + \Delta t$. When velocities are involved in the calculation, use the velocities at moment $t + \frac{\Delta t}{2}$. Then save the new force and torque as $\mathbf{F}(t + \Delta t)$ and $\mathbf{T}(t + \Delta t)$, respectively.

- 4) Update the translational and angular velocity (remaining half-step):

Complete the velocity and angular velocity updates at moment $t + \Delta t$ based on the new force and torque:

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t + \frac{\Delta t}{2}) + \frac{\mathbf{F}(t + \Delta t)}{m} \frac{\Delta t}{2}, \quad (1.40)$$

$$\boldsymbol{\omega}(t + \Delta t) = \boldsymbol{\omega}(t + \frac{\Delta t}{2}) + \frac{\mathbf{T}(t + \Delta t)}{I} \frac{\Delta t}{2}. \quad (1.41)$$

The above procedures will be repeated for every spherical particle in the system until the last time step of the DEM simulation.

1.4.2 DEM models for non-spherical particles

1.4.2.1 Representation of particle orientation

Although using spheres as the shape of solid particles in DEM simulations brings many conveniences, it is undeniable that most solid particles in nature are non-spherical. The proportion of the raw materials consisting of non-spherical particles in modern industries is believed to exceed 70 % [80]. In addition to the obvious difference in packing density between non-spherical and spherical particles, the more significant difference is that the irregular shape of non-spherical particles hinders the overall particle flow during rotational motion. In order to accurately capture the rotation of non-spherical particles in DEM simulation, the orientation of the non-spherical particles must be determined first. This orientation is not only essential in particles' equations of motion, but also crucial for the contact detection.

The representation of particle orientation in DEM is closely related to the moment of inertia. The moment of inertia of a particle is actually a tensor, which can be expressed in the space-fixed (global) coordinate system x - y - z as:

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}, \quad (1.42)$$

where the diagonal elements I_{xx} , I_{yy} and I_{zz} denote the moments of inertia around three axes x , y and z ; the off-diagonal elements are termed the products of inertia, which describe the coupling effect between different axes. By definition, the inertia tensor \mathbf{I} is a symmetric matrix.

For a spherical particle, due to its perfect symmetry, all products of inertia are zero. In addition, its moments of inertia around the three axes are equal, as expressed in Eq. 1.10, and remain constant over time. This is why the moment of inertia of a spherical particle can be simplified as a scalar in Eq. 1.9. However, for a generic non-spherical particle, its products of inertia are not necessarily equal to zero, and the inertia tensor changes in every time step according to the new particle orientation, which makes the calculation of its rotational motion in the space-fixed coordinate system really complex.

Hence, it is convenient to additionally introduce a body-fixed (local) coordinate system that rotates synchronously with the non-spherical particle [81]. According to linear

algebra theory, any real symmetric matrix can be diagonalized through an orthogonal transformation. It means that by rotating the coordinate axes, all the products of inertia in the inertia tensor \mathbf{I} can be zeroed. The axes of the rotated coordinate system are referred to as the principal axes of inertia. The principal axes of inertia can often be identified geometrically, especially for particles with clear symmetry. For particles with uniform density distribution, their symmetry axes are typically their principal axes of inertia. For instance, in the case of elliptical particles, the major and minor axes correspond to the principal axes of inertia. In DEM, the axes of body-fixed coordinate system X - Y - Z coincide with the principal axes of inertia [82], which makes the moment of inertia of the non-spherical particle in the body-fixed coordinate system time-invariant. In this new coordinate system, the inertia tensor is represented as:

$$\mathbf{I}^b = \begin{bmatrix} I_{XX} & 0 & 0 \\ 0 & I_{YY} & 0 \\ 0 & 0 & I_{ZZ} \end{bmatrix}, \quad (1.43)$$

where the diagonal elements of this matrix I_{XX} , I_{YY} and I_{ZZ} are called the principal moments of inertia.

The normalized eigenvectors of \mathbf{I} obtained from this principal axis transformation form the columns of the rotation matrix \mathbf{A} . Two inertia tensors in the space-fixed coordinate system and in the body-fixed coordinate system are connected through the rotation matrix as:

$$\mathbf{I}^b = \mathbf{A}^{-1} \mathbf{I} \mathbf{A}, \quad \mathbf{I} = \mathbf{A} \mathbf{I}^b \mathbf{A}^{-1}. \quad (1.44)$$

Besides, the orthogonality of the rotation matrix, i.e., $\mathbf{A}^{-1} = \mathbf{A}^\top$, enables the accurate transformation of any vector, e.g., velocity or angular velocity, between the two coordinate systems without altering the vector's magnitude or direction [83], which can be written as:

$$\mathbf{v}^b = \mathbf{A}^{-1} \mathbf{v}, \quad \mathbf{v} = \mathbf{A} \mathbf{v}^b, \quad (1.45)$$

where \mathbf{v}^b indicates the vector in the body-fixed coordinate system, whereas the vector in the space-fixed coordinate system has no superscript.

According to Euler's rotation equation, the rotational motion of a non-spherical particle in the body-fixed coordinate system should be expressed as:

$$\mathbf{I}^b \frac{d\boldsymbol{\omega}^b}{dt} + \boldsymbol{\omega}^b \times (\mathbf{I}^b \boldsymbol{\omega}^b) = \sum \mathbf{T}^b, \quad (1.46)$$

or

$$\begin{cases} I_{XX} \frac{d\omega_X^b}{dt} + (I_{ZZ} - I_{YY}) \omega_Y \omega_Z = \sum T_X^b, \\ I_{YY} \frac{d\omega_Y^b}{dt} + (I_{XX} - I_{ZZ}) \omega_Z \omega_X = \sum T_Y^b, \\ I_{ZZ} \frac{d\omega_Z^b}{dt} + (I_{YY} - I_{XX}) \omega_X \omega_Y = \sum T_Z^b. \end{cases} \quad (1.47)$$

Compared to Eq. 1.9, the subscript i , representing a single particle, and the subscript c , representing contact, are omitted for simplicity.

Although the rotation matrix \mathbf{A} can be obtained by diagonalizing the moment of inertia tensor \mathbf{I} in the space-fixed coordinate system, performing this operation at every time step in DEM simulations is computationally expensive and impractical. Therefore, it is necessary to parameterize the rotation matrix to ensure efficient and stable updates at each time step. There are two common parameterization methods: Euler angles and quaternions.

The Euler angles provide an intuitive and easily understandable representation of orientation. They decompose the rotation of the coordinate axes from the space-fixed to the body-fixed system into three sequential rotations. The Euler angles must be combined with a rotation convention, and the most commonly used convention is called Z - X - Z convention [84]. According to this convention, the body-fixed coordinate system X - Y - Z is initially aligned with the space-fixed coordinate system x - y - z . Then, the X - Y - Z system rotates counterclockwise by a precession angle around the original Z -axis. Subsequently, the X - Y - Z system rotates counterclockwise by a nutation angle about the current X -axis. Finally, the X - Y - Z system rotates counterclockwise by a spin angle around the current Z -axis. By these processes, the rotation matrix can be parameterized using the three Euler angles and efficiently updated at each time step in DEM simulations. However, when the nutation angle equals zero or π , the so-called gimbal lock problem arises. Under this condition, a single orientation corresponds to non-unique sets of Euler angles. In other words, when using Euler angles to parameterize the rotation matrix, singularities appear in the update equations for the Euler angles [84]. This is an intrinsic property of the Euler angles and cannot be avoided by selecting other rotation conventions [82], potentially leading to numerical instability in DEM simulations.

Another representation of orientation, quaternions, are singularity-free [85]. The quaternions are interconvertible with Euler angles, but the quaternions corresponding to each orientation are unique. According to Euler's rotation theorem, multiple rotations in three-dimensional space can be expressed as a single rotation around a fixed axis. Let $\mathbf{a} = (a_x, a_y, a_z)$ be the unit axis of rotation from the space-fixed to the body-fixed coordi-

nate system, with a rotation angle β ; the quaternions are then calculated as [60]:

$$\begin{cases} q_0 = \cos \frac{\beta}{2}, \\ q_1 = a_x \sin \frac{\beta}{2}, \\ q_2 = a_y \sin \frac{\beta}{2}, \\ q_3 = a_z \sin \frac{\beta}{2}. \end{cases} \quad (1.48)$$

The quaternions must satisfy the following constraint:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1, \quad (1.49)$$

and the rotation matrix \mathbf{A} can be parameterized by the quaternions as:

$$\mathbf{A} = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix}. \quad (1.50)$$

The rotation matrix \mathbf{A} can then be updated at each time step using the time derivatives of quaternions:

$$\begin{cases} \frac{dq_0}{dt} = \frac{q_1\omega_x + q_2\omega_y + q_3\omega_z}{2}, \\ \frac{dq_1}{dt} = \frac{q_0\omega_x - q_3\omega_y + q_2\omega_z}{2}, \\ \frac{dq_2}{dt} = \frac{q_3\omega_x + q_0\omega_y - q_1\omega_z}{2}, \\ \frac{dq_3}{dt} = \frac{-q_2\omega_x + q_1\omega_y + q_0\omega_z}{2}. \end{cases} \quad (1.51)$$

It is worth noting that any numerical integration algorithm can only provide an approximate solution of Eq. 1.51. Besides, the floating-point arithmetic in computers also causes error accumulation while updating quaternions. For these reasons, the quaternions must be re-normalized in every time step to avoid violating Eq. 1.49.

1.4.2.2 Representation of particle shape

Many models have been developed to represent shapes of non-spherical particles. There are three primary approaches: the polyhedral model, the superquadric model, and the multi-sphere model. Unlike the latter two methods, the polyhedral model is capable of accurately representing particles with edges and flat surfaces, making it more commonly used in geotechnical and geological engineering simulations. However, the complexity of this model is evident, as the types of contact, including vertex-vertex, vertex-edge, vertex-face, edge-edge, edge-face, and face-face, differ significantly. Although various contact detection algorithms have been proposed [86, 87, 88], the calculation of contact forces remains not fully resolved [60]. Overall, the polyhedral model is less popular than

the superquadric model and multi-sphere model. The latter two models are employed in this study and will be described in more detail next.

Superquadric model

The superquadric model uses a continuous function to describe the surface of smooth non-spherical particles. The superquadric equation in the body-fixed coordinate system X - Y - Z given by Barr [89] can be written as:

$$\left(\left| \frac{X}{a} \right|^{n_2} + \left| \frac{Y}{b} \right|^{n_2} \right)^{n_1/n_2} + \left| \frac{Z}{c} \right|^{n_1} - 1 = 0, \quad (1.52)$$

where a , b and c are three shape parameters which indicate the half-lengths of the particle along its principal axes, and n_1 and n_2 are two blockiness parameters. Figure 1.8 gives examples of commonly shaped particles constructed by superquadric equations.

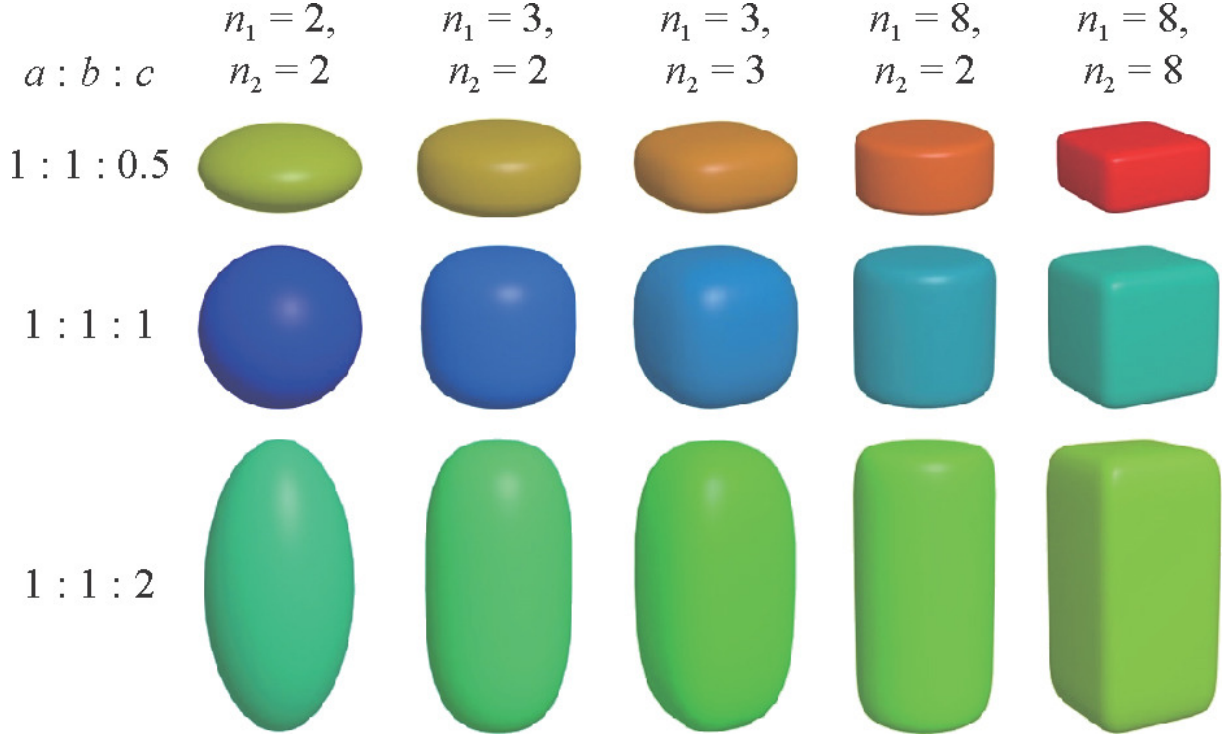


Figure 1.8: Superquadric particle shapes composed by different shape and blockiness parameters [90].

Since the center of gravity of a superquadric particle coincides with its geometric center, its translational motion can also be calculated like that of a spherical particle using Eq. 1.8. Additionally, due to the symmetry of superquadric particles, their principal axes of inertia can be easily determined. Therefore, in DEM simulations, the body-fixed coordinate system can be aligned with the space-fixed coordinate system at the particle initialization stage. This allows the particle orientation to be directly updated using quaternions, beginning with $\mathbf{q} = (1, 0, 0, 0)$, thereby eliminating the step of diagonalizing

the inertia tensor. The analytical expressions of volume and principal moments of inertia of superquadric particles can be found in works by Jaklic et al. [91, 92].

Various numerical integration algorithms [93, 94, 95] can be used to solve Eq. 1.51 in conjunction with Eq. 1.47. The DEM software LIGGGHTS used in this study recommends a novel symplectic quaternion scheme developed by Miller et al [96], which guarantees the energy conservation over long-term DEM simulations. While performing the contact detection, the expressions of different superquadric particles must first be transformed from the body-fixed coordinate system to the space-fixed coordinate system, and then solved by Newton's method. The contact point of two superquadric particles is defined as the equidistant point closest to the surfaces of both particles. The detailed description of particle-particle and particle-wall contact detection algorithms are provided by Podlozhnyuk et al. [60].

Multi-sphere model

The multi-sphere model treats a non-spherical particle as a rigid body composed of multiple primary spheres. The primary spheres are allowed to vary in size and to overlap with each other. This model can approximate non-spherical particles of any shape with adjustable accuracy, and the detailed algorithm can be found in references [97, 98].

The relative positions between primary spheres remain unchanged during the DEM simulations. At each time step, the total force and torque of the multi-sphere particle are the sum of the contributions from all primary spheres and are accumulated at its center of gravity. The translational and rotational motion of the multi-sphere particle is then updated based on these resultant forces and torques. Note that the total mass of the multi-sphere particle cannot be simply calculated as the sum of the masses of all primary spheres due to the potential overlapping between primary spheres. Actually, the mass, the center of gravity and the inertia tensor of the multi-sphere particle are all accessed by a Monte Carlo procedure in LIGGGHTS. A multi-sphere particle does not necessarily exhibit symmetry, and its center of gravity does not always coincide with its geometric center. So the inertia tensor of the multi-sphere particle needs to be diagonalized during the particle initialization stage to determine the rotation matrix. The initial quaternions can then be expressed using the elements of the rotation matrix \mathbf{A} as:

$$\begin{cases} q_0 = \frac{1}{2}\sqrt{1 + A_{11} + A_{22} + A_{33}}, \\ q_1 = \frac{1}{4q_0}(A_{32} - A_{23}), \\ q_2 = \frac{1}{4q_0}(A_{13} - A_{31}), \\ q_3 = \frac{1}{4q_0}(A_{21} - A_{12}). \end{cases} \quad (1.53)$$

The numerical integration algorithm in the multi-sphere model is similar to the velocity

Verlet integration for spherical particles, but includes an additional error correction procedure using the Richardson extrapolation to further enhance the accuracy of quaternion calculations at each time step. The contact detection of multi-sphere particles is essentially the determination of the contact between primary spheres. So the fast and robust contact detection algorithm for spherical particles can be easily extended to the multi-sphere model. This convenience, as well as its flexibility, make the multi-sphere model the most popular approach for representing non-spherical particles [82].

1.4.3 Determination of DEM parameters

The determination of DEM parameters is a critical step in ensuring both the efficiency and reliability of simulations. There are numerous input parameters in DEM simulations, which can be broadly classified into two main types: simulation parameters and physical properties. Simulation parameters, including time step and skin distance, primarily affect the CPU time of running a DEM simulation. Improper settings in this category can lead to excessive computational demands or even numerical instabilities. In contrast, physical properties, such as Young's modulus, Poisson's ratio and coefficient of friction, define the interaction laws used in the DEM model. Accurate calibration of physical properties is essential for ensuring that simulation results are consistent with experimental observations and physical reality. Although these two categories serve distinct purposes, they are also interconnected, which will be discussed in detail next.

1.4.3.1 Simulation parameters

Time step

The choice of time step is of great importance in DEM simulations. The time step must be set sufficiently small for two main reasons. First, it prevents excessive overlaps between contacting particle pairs, which may lead to unrealistically high contact forces. This reason is well understood and is demonstrated in Eq. 1.39, where an excessively large time step results in unphysically large particle displacements in the subsequent iteration. Second, small time step can avoid effects of disturbance waves. The movement of a particle in a granular flow is affected not only by the forces and torques originated from its immediate neighboring particles but also the particles far away through the propagation of disturbance waves [99]. By choosing a small enough time step in DEM, disturbance waves from each particle are prevented from propagating further than to its neighboring partners. It is assumed that all of the radiated energy in disturbance waves is transferred by Rayleigh waves, therefore the contributions of distortional and dilational waves are neglected [100].

The critical time step (also referred to as Rayleigh time step) is calculated based on the

theoretical expression of the Rayleigh wave velocity and can be approximately expressed as [101]:

$$\Delta t_{crit} = \frac{\pi r \sqrt{\frac{\rho}{G}}}{0.1631\nu + 0.8766}, \quad (1.54)$$

where r is the particle radius, ρ is the particle density, G is the shear modulus of the particle and ν is Poisson's ratio of the particle. Due to the involvement of spherical and non-spherical particles with varying sizes and densities in this study, the most conservative values are adopted for particle radius and density: a radius of 1.5 mm for the smallest spherical particle and a density of 1,000 kg/m³ for the lightest particle used. The shear modulus G is related with the Young's modulus E and Poisson's ratio ν :

$$G = \frac{E}{2(1 + \nu)}. \quad (1.55)$$

The Young's modulus of the particle in this study is set as 1×10^7 Pa and the Poisson's ratio is 0.24. Therefore, the critical time step yields $\Delta t_{crit} = 8.2 \times 10^{-5}$ s. Note that Eq. 1.54 assumes that the relative velocity between contacting particles is very small. Other than for quasi-static systems, the time step in DEM simulations should be set as a fraction of the critical time step, typically ranging from 20% to 80% in practice [27]. The time step of all the DEM simulations in this study is finally determined as $\Delta t = 2 \times 10^{-5}$ s, which is 24% of the critical time step.

Skin distance

Although the criterion of contact between two spherical particles is the simplest (Eq. 1.12), it still involves calculating the distance between the particle pair. Computing the distances for all particle pairs in the system at every time step would result in an enormous computational burden and is unnecessary. Therefore, optimizing the computational strategy for contact detection is critical. In LIGGGHTS, this optimization is achieved through the use of neighbor lists, a technique that is equally applicable to non-spherical particles [60]. Before performing contact detection, the neighbor list excludes particle pairs that are a priori too far apart to interact within several time steps. This technique significantly reduces the number of particle pairs requiring distance checking, thereby greatly improving the computational efficiency of DEM simulations, especially for systems with large particle counts.

The neighbor list is constructed through the combination of particle binning [102, 103] with the Verlet list [78]. Particle binning implies that the simulation domain is divided by equal-sized cubic bins, with all particles assigned to corresponding bins according to their center coordinates. This approach facilitates the subsequent identification of neighbors by bypassing direct pairwise distance calculations. The Verlet list determines the range for identifying neighboring particles and the frequency of neighbor list updates. Specifically,

the neighbor search radius r_{neigh} consists of the force cutoff distance r_f and an additional buffer region known as the skin distance r_s , which can be written as:

$$r_{neigh} = r_f + r_s. \quad (1.56)$$

For spherical particle pair of identical size, the force cutoff distance is equal to twice of the particle radius. In contrast, for more general cases involving non-spherical particles of varying sizes, the force cutoff distance is defined as the maximum possible center-to-center distance at which two particle surfaces may come into contact. The skin distance, which is a user-specified value, determines when the neighbor list is updated. An update occurs only when any particle in the system moves more than half of the skin distance within a time step [64]. On the one hand, this rule allows the neighbor list to be reused over multiple time steps, thereby improving the efficiency of contact detection. On the other hand, it ensures that no potential neighbors newly entering the search radius are missed during contact detection at any time step. The bin size (the length of the cubic side) is typically matched to the neighbor search radius. In LIGGGHTS, the default setting for the bin size is half of the maximum neighbor search radius in the system, as this configuration has been found to be optimal for many typical cases [65].

Figure 1.9 illustrates an example of neighbor identification in a 2D simulation domain. In this system, circular particles with identical radii are assumed. The skin distance is set equal to the particle radius in this case. However, regardless of the skin distance value, as long as the bin size s_{bin} is set to half of the neighbor search radius r_{neigh} , considering all particles located within a range of 25 (5×5) bins around a given particle ensures that no neighbors are missed. Extending this method to a 3D simulation domain, the neighbor search requires considering only 125 ($5 \times 5 \times 5$) bins around each particle, and no distance calculations are needed during this neighbor identification process.

Although the size of the skin distance does not affect the total number of bins considered during the neighbor search, it plays a critical role in the overall process of neighbor list construction and subsequent contact detection. When the skin distance is small, the number of neighbors, and thus the number of distance calculations required during contact detection, is reduced. However, the neighbor list must be updated more frequently, and the increased number of bins in the simulation domain results in higher memory requirements for storing the neighbor list. Conversely, when the skin distance is large, the opposite occurs: fewer neighbor list updates are needed, but the number of distance calculations increases. Therefore, the optimal value of the skin distance is typically determined through trial simulations. Table 1.2 gives the results of trial simulations for determining the value of skin distance. It was found that setting the skin distance to one particle radius resulted in the shortest DEM simulation time. Therefore, a skin distance of 1.5 mm was adopted for all simulations in this study. It is worth noting that a smaller skin distance does not

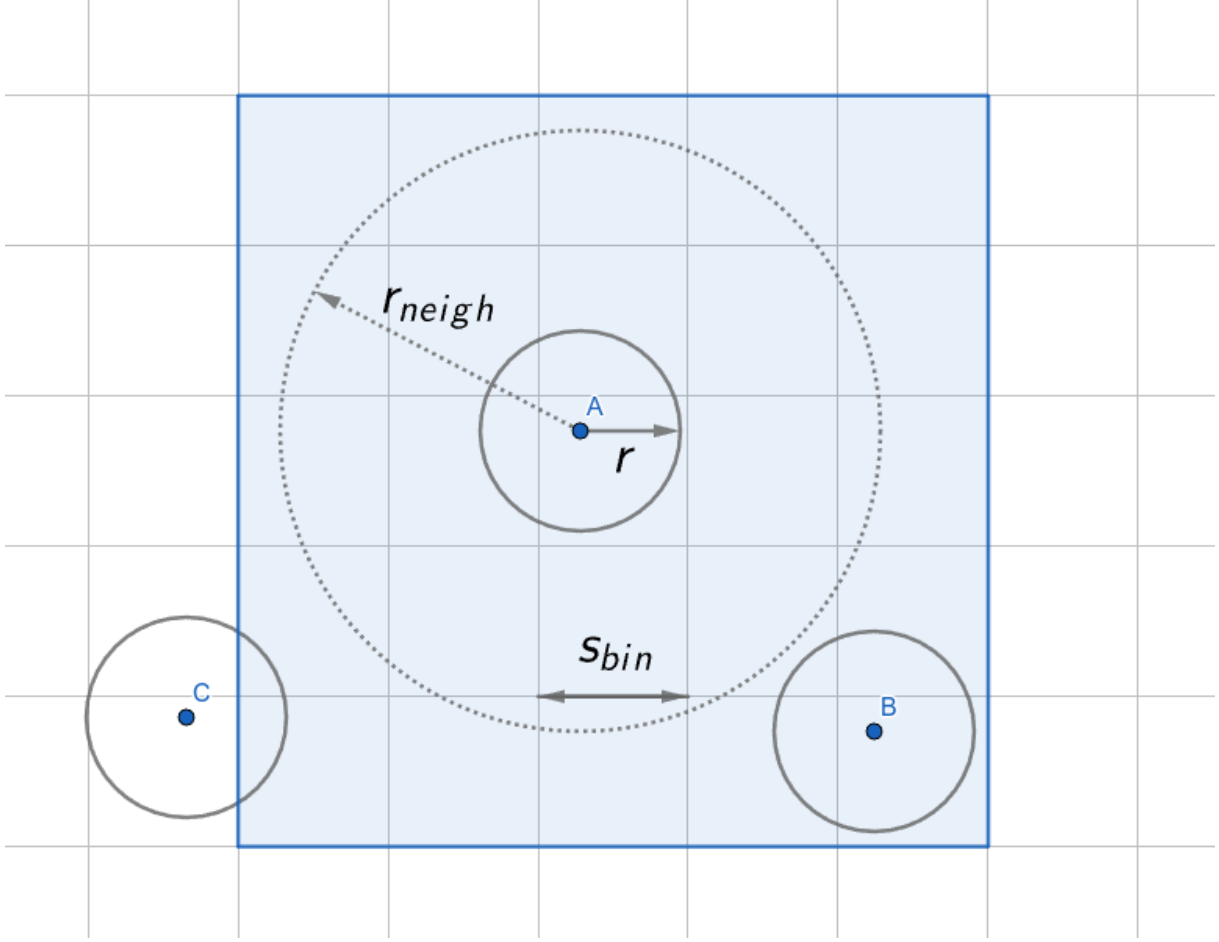


Figure 1.9: Schematic representation of neighbor search in a 2D simulation domain, $r_{neigh} = r_f + r_s = 2r + r = 3r$, $s_{bin} = 0.5r_{neigh} = 1.5r$. Particle B is a neighbor of particle A, while particle C is not.

always lead to reduced simulation time in other DEM simulations. The determination of the optimal skin distance always requires targeted trial simulations. The trends of CPU time presented in Table 1.2 may be attributed to the relatively small drum size (simulation domain) used in this study, where an increase in the total number of bins has a negligible impact on simulation efficiency.

Table 1.2: Comparison of trial simulations with different skin distances (37,600 spherical particles of 1.5 mm radius mixed in a rotary drum for 1 s).

Skin distance [mm]	Average neighbors per particle [-]	CPU time [s]
1.5 (r)	6.8	208
3.0 ($2r$)	17.9	286
4.5 ($3r$)	33.1	426
6.0 ($4r$)	58.7	626

1.4.3.2 Physical properties

In this study, there are five physical properties that remain constant throughout all DEM simulations: Young's modulus, Poisson's ratio, coefficient of restitution, rolling resistance coefficient, and coefficient of friction. For common experimental materials, the values of the first three physical properties can be easily found in the literature or reference books. Among them, it is worth noting the setting of Young's modulus. Taking the experimental material γ -alumina as an example, its actual Young's modulus is approximately 4×10^{11} [104], which leads to excessively small time step (Eqs. 1.54 and 1.55) as well as large computational time. To overcome this limitation, it is quite common in the literature to reduce the value of the Young's modulus when performing DEM simulations [105]. Another common simplification is to use the same properties for wall and particle (as well as for particle-particle and particle-wall interactions), mainly due to the fact that walls do not undergo any deformation in DEM simulations. The simulation results from Yamamoto et al. [106] also indicated that Young's modulus, Poisson's ratio and coefficient of restitution are not significant for particle motion.

As for the other two friction-related coefficients, they are often difficult to obtain from the literature, and direct measurement is equally challenging. This is not a concern in DEM practice, as neither of these two approaches is generally recommended. The rolling resistance coefficient for stiff sphere, such as γ -alumina particle in this study, is often set to a considerably small value or even ignored [107, 108]. The coefficient of friction has the greatest effect on particle motion among those five physical properties. However, even if the coefficient of friction is accurately measured, it cannot be directly used in DEM simulations. Because it needs to compensate for the effects of using a reduced Young's modulus [109]. In practice, the coefficient of friction used in DEM is typically calibrated by comparing batch simulation results with the experiment. And it is acceptable to assign different coefficients of friction for particle-particle and particle-wall interactions.

In this study, the calibration target is the dynamic repose angle, as already introduced in Section 1.3.1 and illustrated in Fig. 1.5. The strategy begins by simplifying the calibration process by setting identical coefficients of friction for particle-particle and particle-wall. Using the calibrated coefficient of friction, a full-scale simulation was performed and validated by quantitatively comparing the simulation results with experimental data through the selected mixing index. If validation failed, different values of coefficient of friction for particle-particle and particle-wall contacts would be explored to achieve a better match. Fortunately, the identical setting of both coefficients of friction proved sufficient to satisfy the validation criteria in this study. The mixing indices will be detailed in Section 1.5.1. The specific calibration and validation procedures are described in Chapter 2 (the first manuscript), along with the numerical values of all physical properties, which will not be repeated here.

1.5 Mixing quantification

In DEM simulations, instantaneous dynamic information about particles, such as position, velocity, and trajectory, can be readily obtained, whereas experimental measurements are often more challenging. To gain deeper insights into particle mixing in rotary drums, numerous experimental techniques have been developed. One of the earliest approaches involved direct sampling methods, which, despite their effectiveness in reflecting local mixing conditions, are inherently invasive and can disrupt the overall mixing state of the system. To address this limitation, non-invasive techniques such as radioactive particle tracking (RPT) [110, 111], positron emission particle tracking (PEPT) [112], magnetic particle tracking (MPT) [113] and magnetic resonance imaging (MRI) [114, 115] have been introduced. The first three particle tracking methods allow for measuring a limited number of tracer particles, which is valuable for the microscopic analysis of granular flow. However, they do not provide information for the majority of ordinary particles in the system. As for the MRI, while capable of visualizing the internal structure of particle beds, it is limited by the high equipment costs and stringent requirements for the experimental particles, as only a small selection of MRI-sensitive materials can be used. In contrast, image analysis methods have gained widespread application due to their simplicity, low experimental cost, and flexibility in image processing algorithms. Although image analysis is restricted to capturing particle information from the front and rear walls of transparent drums, it is sufficient for validating the reliability of DEM simulations. Therefore, image analysis was adopted as the primary experimental technique in this study. The detailed validation procedure and image processing methodology are presented in Chapter 2 (the first manuscript).

1.5.1 Mixing indices

A unified quantitative description of mixing time and mixing quality is essential for both experimental validation of DEM parameters and the analysis of subsequent DEM mixing simulations. To achieve such standardization, various mixing indices have been proposed in the literature. According to the principle of calculation, those mixing indices can be divided into two categories: fraction-based and distance-based. Fraction-based indices quantify the number fraction of different particle types within selected regions, with typical examples including the Lacey index [1], the entropy of mixing [116], and the subdomain-based mixing index [117]. In general, these indices range from zero to one, where zero represents complete segregation and one indicates perfect mixing. By definition, fraction-based indices are inherently influenced by the size of the selected regions. This can be intuitively understood: if a region encompasses the entire system,

the local particle fraction equals the overall composition, resulting in a mixing index of one. Conversely, if the regions are too small, each containing only a single particle or none, then every local mixing index equals zero, leading to a global mixing index of zero. Therefore, the choice of region size is critical. Regions that are too small not only underestimate the overall homogeneity but also increase the total number of regions, raising computational costs. On the other hand, excessively large regions fail to capture local heterogeneity, which may prevent the index from accurately identifying segregation in rotary drums. Distance-based indices, in contrast, evaluate particle spatial distribution by computing relative positions through coordinate analysis. Common methods include the nearest-neighbors and neighbor-distance approaches [118]. Unlike fraction-based indices, distance-based indices consider all particles in the system without relying on selected regions, thus eliminating the influence of region size. Nevertheless, distance-based indices are obviously more computationally expensive and generally applicable only to small-scale mixing systems with fewer than 10,000 particles. Given that the particle count in this study reaches up to 75,200, one of the three fraction-based mixing indices was selected for further mixing analysis, which will be introduced in detail in the following subsections.

1.5.1.1 Lacey index

The Lacey index is one of the most famous and classic mixing indices for analyzing the mixing state of binary mixtures. It is based on the statistics and can be defined as:

$$LI = \frac{\sigma_u^2 - \sigma^2}{\sigma_u^2 - \sigma_m^2}, \quad (1.57)$$

where σ^2 is the variance of number fraction of a specific kind of particles, σ_u^2 is the variance in completely unmixed state and σ_m^2 is the variance in fully mixed state. These three variances can be expressed as:

$$\sigma^2 = \frac{1}{N_s} \sum_{i=1}^{N_s} (p_i - p)^2, \quad (1.58)$$

$$\sigma_u^2 = p(1 - p), \quad (1.59)$$

$$\sigma_m^2 = \frac{p(1 - p)}{N_p}, \quad (1.60)$$

where N_s is the number of selected regions, N_p is the number of particles in each of the selected regions, p_i is the number fraction of the specific kind of particles in the region i , and p is the number fraction of the specific particles in the whole mixture. By definition, N_p should be a constant, which implies that this index requires an identical amount of particles in each of the selected regions. This condition is hard to achieve in practice,

so that N_p is usually replaced by the average number of particles in all selected regions. However, this replacement may lead to a smaller N_p , which in turn can make the Lacey index in Eq. 1.57 to exceed one. This restriction about N_p also prevents the Lacey index from being applied through grid division of the entire particle system, making it highly sensitive to the location of the selected regions. This effect becomes particularly pronounced when particle segregation occurs. Moreover, because Eqs. 1.59 and 1.60 are only valid for binary mixtures, the Lacey index is not able to characterize the mixing state of multi-component mixtures.

1.5.1.2 Entropy of mixing

The entropy of mixing has become a popular mixing index in recent years. Unlike the Lacey index, it does not require an equal number of particles in each selected region, therefore allowing for grid division of the particle system and consideration of all particles within the system. It is important to note that, despite sharing the same name as the thermodynamic concept, this index does not carry the same physical meaning; rather, it is only a mixing measure inspired by the mathematical form of thermodynamic entropy. For a binary mixture, the entropy of the whole mixing system is expressed as:

$$S = p_1 \ln p_1 + p_2 \ln p_2, \quad (1.61)$$

subject to:

$$p_1 + p_2 = 1,$$

where p_1 and p_2 are the initial number fractions of the two types of particles in the mixing system. The local entropy of mixing in a specific grid element i is similarly calculated as:

$$S_i = p_{1,i} \ln p_{1,i} + p_{2,i} \ln p_{2,i}, \quad (1.62)$$

where $p_{1,i}$ and $p_{2,i}$ are the number fractions of the two types of particles in the grid element i . Subsequently, all of the local mixing indices $\frac{S_i}{S}$ are weighted and averaged in N_s grid elements according to the number of particles to obtain the total mixing index EI as:

$$\text{EI} = \sum_{i=1}^{N_s} \frac{S_i}{S} \frac{N_{p,i}}{N_{p,tot}}, \quad (1.63)$$

where $N_{p,i}$ is the number of particles in the grid element i , and $N_{p,tot}$ is the total number of particles in the whole mixing system. This algorithm enables the analysis of every local region within the mixing system. In addition, according to the expression of entropy (Eq. 1.61), this mixing index can be easily extended from binary to multi-component mixtures. Both of these features are not achievable with the Lacey index. However, the extreme

value of S in Eq. 1.61 can only be achieved when $p_1 = p_2 = 0.5$, which means the local mixing index is guaranteed to be less than one only when the two particle types have equal initial proportions in the system. The same limitation applies when the entropy of mixing is used for multi-component mixtures, where the local index remains valid only if all particle types have identical initial proportions within the system.

1.5.1.3 Subdomain-based mixing index (SMI)

The subdomain-based mixing index (SMI) is a recently proposed mixing index that overcomes all the limitations of the previously mentioned Lacey index and entropy of mixing. The core concept of SMI is similar to that of the entropy of mixing, as it also partitions the entire particle system into multiple subdomains through grid division and integrates local indices to obtain the total mixing index. For a mixture that consists of Q types of particles within N_s subdomains, the participation factor of k -type particles ($1 \leq k \leq Q$) can be calculated as:

$$f_k = \frac{\max \left(\sum_{i=1}^{N_s} N_{p,1i}, \sum_{i=1}^{N_s} N_{p,2i}, \dots, \sum_{i=1}^{N_s} N_{p,Qi} \right)}{\sum_{i=1}^{N_s} N_{p,ki}}, \quad (1.64)$$

where $N_{p,ki}$ is the number of k -type particles in subdomain i . Therefore, the participation factor f_k represents the ratio of the total number of the most abundant particle type in the system to the total number of k -type particles. The modified fraction of k -type particles in subdomain i is calculated as:

$$P_{ki} = \frac{N_{p,ki} f_k}{\max (N_{p,1i} f_1, N_{p,2i} f_2, \dots, N_{p,Qi} f_Q)} \leq 1. \quad (1.65)$$

If all types of particles are perfectly mixed in subdomain i , it means that the proportion of each particle type within subdomain i is identical to its original proportion in the entire system, therefore $P_{1i} = P_{2i} = \dots = P_{ki} = \dots = P_{Qi} = 1$ and $\sum_{k=1}^Q P_{ki} = Q$. In contrast, if complete segregation occurs in subdomain i , for example, only k -type particles exist in subdomain i , then $\sum_{k=1}^Q P_{ki} = P_{ki} = 1$. Then, the local SMI for subdomain i is the normalization of $\sum_{k=1}^Q P_{ki}$ from the range of $[1, Q]$ to the range of $[0, 1]$ as:

$$\text{SMI}_i = \frac{1}{Q-1} \left(\sum_{k=1}^Q P_{ki} - 1 \right). \quad (1.66)$$

In addition, if a subdomain does not contain any particles, the local SMI is directly assigned a value of zero, which similarly represents complete segregation. Finally, the total

mixing index is obtained by averaging all the local indices as follows:

$$\text{SMI} = \frac{1}{N_{p,tot}} \sum_{i=1}^{N_s} \left[\text{SMI}_i \sum_{k=1}^Q N_{p,ki} \right], \quad (1.67)$$

where $N_{p,tot}$ is the total number of particles in the whole mixing system.

Through sophisticated algorithm design, the SMI not only guarantees the strict accuracy of local indices in any subdomain but also remains applicable to multi-component mixtures with varying initial proportions. Its capabilities allow for a comprehensive evaluation of both local and overall mixing under complex mixing conditions. Consequently, it is selected in this study as the sole mixing index for quantitatively describing particle mixing in rotary drums.

1.6 Motivation and objectives

So far, the fundamental aspects of solid particle mixing in rotary drums, the numerical simulation method and the quantitative description of mixing process have been covered in detail. The particle motion in rotary drum is quite complex, particularly in the presence of segregation. Various factors influence particle mixing in rotary drums, which can be broadly classified into two categories: drum-related parameters such as drum size, revolution frequency and filling degree; and particle-related properties, with particle shape, size ratio and density ratio being the most critical. The interplay of these factors leads to distinct particle motions, and thus significantly affects the mixing efficiency and mixing quality. Existing studies have primarily focused on the influence of individual parameters, and are mostly qualitative investigations. However, systematic quantitative research that simultaneously considers multiple influencing factors is scarce. The ability to predict mixing outcomes across a broad parameter space remains an open question. While the discrete element method (DEM) has been widely employed for simulating particle mixing, its computational costs impose significant limitations, particularly for large-scale particle systems or those involving non-spherical particles.

Therefore, in order to address these gaps, the central goal of the present work is to efficiently predict the particle mixing results in rotary drums within a parameter space that encompasses multiple influencing factors, based on a limited number of DEM simulations. This cumulative dissertation comprises three published manuscripts as Chapters 2 to 4. Chapter 2 explores a novel approach for predicting the mixing time of spherical particles in a rotary drum, termed cross-correlation. This method establishes a bridge between 2D and 3D simulations, enabling efficient predictions of 3D mixing time across different rotational speeds and density ratios using cost-effective 2D simulations. Chapter 3 presents a machine learning model: particle swarm optimized support vector regression (PSO-SVR)

to fast predict mixing of spherical particles in rotary drums. This model not only enables accurate predictions of both the mixing time and the mixing degree at steady mixing state but also expands the considered influencing factors to four: revolution frequency, size ratio, density ratio, and drum length. Chapter 4 extends the application of machine learning methods to predictions for a non-spherical particle type—rod-like particles. This chapter also provides a detailed comparison of three different machine learning models, covering both their underlying principles and predictive performance. Chapter 5 summarizes the main conclusions from the three manuscripts and provides an outlook on possible further work.

Chapter 2

Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation

During the mixing of solid particles in a rotary drum, particle motion in the transverse plane is significantly more intense than in the axial direction. Building on this observation, the feasibility of using 2D simulations as a cheap surrogate for computationally expensive 3D simulations was explored. This approach enabled the rapid prediction of mixing time for spherical particles in a drum under varying revolution frequencies and density ratios.

A paper with the same title as this chapter has already been published:

Wencong Wu, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation.” *Advanced Powder Technology* 33.4 (2022): 103512.

As this paper is not open access, the attached version is the preprint manuscript containing the same content as the published version. For highlights and the graphical abstract, please visit:

<https://doi.org/10.1016/j.appt.2022.103512>.

The validation experiment in this paper was conducted by my mentor and second author Dr.-Ing. Kaicheng Chen. As the first author, my **individual contributions** include: designing and performing all DEM simulations, analyzing the simulation data, developing the cross-correlation for mixing prediction, and writing the original draft with all figures and tables.

Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation

W. Wu*, K. Chen, E. Tsotsas

*Thermal Process Engineering, Otto-von-Guericke-University, Universitätsplatz 2, 39106
Magdeburg, Germany*

Abstract

A new concept for the prediction of the mixing time of a large particle system from the mixing time of a small particle system and a scale-spanning cross-correlation are presented. By way of example, the considered large system is a 3D rotary drum, and the used small system is a 2D rotary drum. At both levels, data for the change in mixing degree with time are created by DEM simulation. The cross-correlation is developed from a learning set which consists of 17 sets of simulation data and represents a variation of revolution frequency. The prediction of mixing time through the cross-correlation is excellent within the parametric range of the learning set ($R^2 = 0.92$). Beyond the parametric range of the learning set, good predictions are obtained for binary mixtures of particles with different density ($R^2 = 0.86$), but not in case of different size. Whether the additional parameter will affect the axial mixing in the rotary drum is considered to be the key for such parameter expansion. Advantages of interpretability, expandability and relatively high accuracy for the limited data size compared to machine learning approaches

*Corresponding author. Tel.: +49 391 67 52322.
Email address: wencong.wu@ovgu.de (W. Wu)

are seen in the cross-correlation method, which may promote its application for the fast and cheap prediction of industrial mixers.

Keywords: Particle mixing, Rotary drum, DEM simulation, Prediction, Machine learning

1. Introduction

Mixing is a standard unit operation for particles in chemical, food, feed and pharmaceutical industries. An ideal mixer should yield a uniform mixture in relatively short time, which reveals the two worth noting aspects of a mixing process: the mixing degree and the mixing time. The mixing degree is a quantitative description of the uniformity of a mixture, which clearly reflects the final product quality. And the mixing time directly determines the efficiency of a mixing process.

To improve the performance of mixers or to assess the behavior of new products in such equipment, computer modeling is an economical and convenient choice. Since the discrete element method (DEM) has been introduced by [Cundall and Strack \(1979\)](#), it has been widely used for particle-related simulations, including simulations of mixing. However, even in recent publications, most mixing simulations with DEM are still restricted to small-scale systems with total particle numbers less than 20000 ([Liu et al. \(2017\)](#); [Chen et al. \(2017\)](#); [Zhang et al. \(2020\)](#); [Hlostá et al. \(2020\)](#)). This is because 3D DEM simulation is computationally expensive. Simulations from [Bertrand et al. \(2005\)](#) show that the CPU time increases exponentially with the number of particles in the mixing system. Although DEM modeling with millions of particles can be achieved through supercomputers with clusters of CPU or

GPU nowadays ([Govender et al. \(2018\)](#); [Yan and Regueiro \(2018\)](#)), the particle number is still far away from a typical industrial mixing system, which usually includes billions of particles.

This situation underlines the still persisting need for methods that would provide access to the mixing dynamics of large particle systems at low cost. As it will be discussed later on, machine learning or other similar techniques could be used to this purpose. In contrast, a different approach is here proposed. The new idea is to simulate a small particle system per DEM and then connect the result of such simulations by means of a cross-correlation with the mixing behavior of a large scale particle system. Learning would in this concept be top-down (derivation of the cross-correlation by comparison of the large with the small system), application would be bottom-up (predictions for the large system from small system simulations and the previously derived cross-correlation). In the present work, the large scale particle system is represented by 3D DEM simulations for a rotary drum, whereas a flat, 2D rotary drum is the small, surrogate particle system, simulated by 2D DEM. In general, however, any mixer could be the large system, with either experimental or computational data for the representation of its mixing behavior. And, other surrogates are thinkable, meaning that large mixer and surrogate don't necessarily need to be of similar geometry.

In any case, questions that arise are whether the proposed approach is at all possible, i.e. whether such a link between large and surrogate system can be expressed in form of a cross-correlation, which accuracy can be achieved in this way, and how robust the link would be, i.e. when the surrogate representation can be expected to work or fail. These aspects will be treated

in the paper for the already mentioned case of 3D rotary drum DEM as the large particle system, and 2D rotary drum DEM as the computationally cheap surrogate. We are not aiming at fundamentally new insights about the dynamics of particles in rotary drums, but tear profit from the fact that rotary drums are a simple, commonly used, and well understood example of a mixer.

In a rotary drum, particles are mixed in both transverse plane and axial direction. The axial mixing is really slow and considered as driven by self-diffusion ([Finnie et al. \(2005\)](#)). The transverse mixing is a few orders of magnitude faster compared to mixing in the axial direction ([Nityanand et al. \(1986\)](#)), which makes 2D DEM simulations with a small amount of particles that can significantly reduce the computational cost especially interesting. [Kwapinska et al. \(2006\)](#) compared 2D DEM simulation results for a rotary drum with experiments and penetration models for thermal processes. [Xu et al. \(2010\)](#) conducted a qualitative parameter study through 2D DEM simulations. However, 2D simulation has always been treated as a simplified model to qualitatively analyze the dynamics of the mixing system. The use of 2D simulations as a cheap surrogate for mixing prediction is basically unexplored.

The paper is organized as follows. In Section 2, it is explained how mixing time is derived from transients of an appropriate mixing index obtained from either 2D or 3D simulations, including experimental validation of the latter. In Section 3, data are separated into a learning set and several prediction sets, similarly as in machine learning approaches. The learning set takes the widely investigated influential parameter revolution frequency as the unique

variable to develop the cross-correlation. The predictive ability of the developed cross-correlation is then tested within and, in several steps, also beyond the parametric range of the learning set. A short discussion that compares the new cross-correlation method with prediction models from the field of machine learning in regard of notable merits and future potential is given in Section 4, followed by conclusions in Section 5 and some supplementary material in three Appendices. It should be noted that, though steady-state mixing quality is discussed in some cases, mixing time is the target quantity of this work, so that the cross-correlation and respective analysis refer only to mixing time.

2. Experiment and DEM modeling

2.1. *Mixing experiment*

Though not absolutely necessary from the theoretical point of view (inaccurate or partially biased data might also be used, as long as the main features of mixing dynamics of the large system are still reflected), reliability and accuracy of the 3D DEM simulation would be an advantage in the present work.

Hence, a lab scale experiment was carried out for calibration and validation. The experimental setup consists of a horizontal rotary drum mixer and a camera system, as illustrated in Fig. 1(a). The rotary drum is made of acrylic glass with an inner diameter of 194 mm and a length of 150 mm. The entire drum is transparent with a black background at the rear face, which facilitates imaging. The camera system includes a LaVision high-speed camera with CMOS chip (1440×1440 pixels) and two halogen lamps (400 W)

providing continuous lighting. Mixing material is γ -alumina spherical particles with diameter of 3 mm and particle density of 1358.8 kg/m^3 . In order to create a binary mixture, half of the particles were left with their original white color, and another half were dyed light green in a spray fluidized bed. Before starting the experiment, particles of the two colors, each with the same amount, were placed in the drum in top-bottom stratification (Fig. 1(b)), at a filling degree of the drum of 20%. With a set value of 15 rpm for the revolution frequency of the drum, the entire mixing process lasted for 60 s. During this time, images from the front face of the drum were captured by the high speed camera with 40 frames per second. After the experiment, image processing is necessary to extract the information on particle positions from the raw images for further analysis, more details are shown in [Appendix A](#).

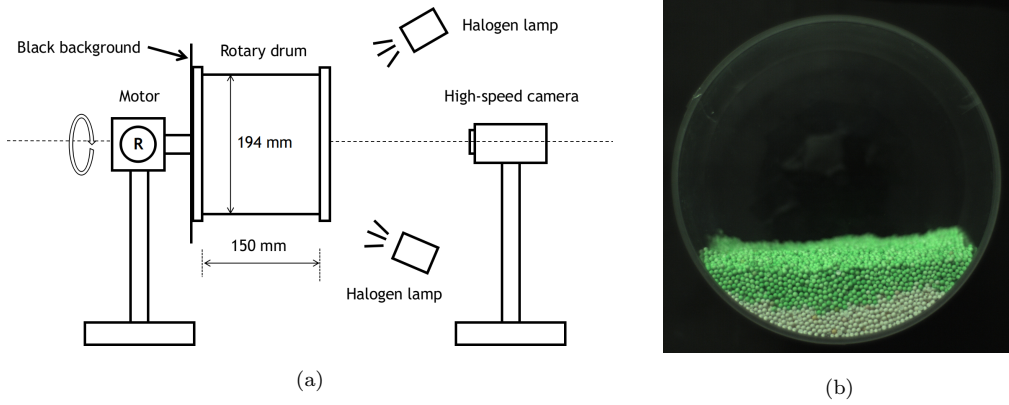


Figure 1: Experimental system: (a) sketch of setup (b) rotary drum with top-bottom stratification.

2.2. DEM modeling and validation

The open-source software LIGGGHTS (Kloss et al. (2012)) was used to conduct the DEM simulations in the present work. Governing equations can be found in Appendix B. To accurately simulate the mixing experiment, physical properties such as Young’s modulus, Poisson’s ratio, coefficient of restitution, rolling coefficient and friction coefficient need to be carefully determined (see Appendix C). Respective values are summarized in Table 1. In this study, physical properties were assumed identical for particles and walls.

Table 1: 3D DEM simulation parameters.

Parameters	Value	Unit
Time step	2×10^{-5}	s
Young’s modulus	1×10^7	Pa
Poisson’s ratio	0.24	-
Coefficient of restitution	0.82	-
Rolling coefficient	0.01	-
Friction coefficient	0.80	-

After parameter determination, mixedness comparison between experiment and simulation is necessary for further validation. In order to describe the quality of mixing quantitatively, different mixing indices were developed in the literature (Lacey (1954); Schutyser et al. (2001); Deen et al. (2010)). Our mixing system contains 37600 particles. To avoid the bias from randomly selected samples and considering the computational cost, a recently proposed index named subdomain-based mixing index (SMI) (Cho et al. (2017)) was

selected as the measure of mixedness in this study. This index takes all the particles in the mixing system into account, and can be calculated by means of a relatively simple algorithm. Determination of the SMI starts by dividing the computational domain into n subdomains. For a mixture consists of Q types of particles, the participation factor of particle type k can be calculated as:

$$f_k = \frac{\max(\sum_{i=1}^n N_{1i}, \sum_{i=1}^n N_{2i}, \dots, \sum_{i=1}^n N_{Qi})}{\sum_{i=1}^n N_{ki}}. \quad (1)$$

The modified fraction of k -type particles with respect to the maximum number of particles among the Q types in subdomain i is:

$$P_{ki} = \frac{N_{ki}f_k}{\max(N_{1i}f_1, N_{2i}f_2, \dots, N_{Qi}f_Q)}. \quad (2)$$

The local SMI for subdomain i is calculated from the arithmetic mean of fraction P_{ki} over all particle types, except of the majority type of particles with a fraction which by definition is equal to one:

$$\text{SMI}(i) = \frac{1}{Q-1} \left(\sum_{k=1}^Q P_{ki} - 1 \right). \quad (3)$$

Finally, the total mixing index is obtained by averaging all the local SMIs as follows:

$$\text{SMI} = \frac{1}{N_{tot}} \sum_{i=1}^n \left[\text{SMI}(i) \sum_{k=1}^Q N_{ki} \right], \quad (4)$$

where N_{tot} is the total number of particles in the whole mixing system. The range of SMI is between zero and one: zero means that different particles are fully separated, whereas one indicates a perfect mixing state.

For a mixing process, two characteristics of mixedness are noteworthy: mixing time and mixing degree at steady state. Mixing time is the time after

which the mixing process can be assumed to have reached a steady mixing state, so that the mixing degree of the mixture will only fluctuate around a certain value. In order to describe these two characteristics quantitatively, the curve that describes the transient of SMI over time can be fitted by the following equation:

$$\text{SMI} = ae^{-bt} + c. \quad (5)$$

Here c is the asymptote of the fitted curve, which can be treated as the mixing degree at steady mixing state. And, the moment when the mixing index becomes equal to $0.95c$ is defined as the mixing time.

Subdomain-based mixing index was tested by the 3D benchmark simulation, which has the same particle configuration and operating conditions as in the experiment. As remarked by [Cho et al. \(2017\)](#), the total number of subdomains for a mixing system has an influence on the value of mixing index. Larger number of subdomains leads to a lower value of mixing degree at steady mixing state, as can be seen in Fig. 2(a). Considering the time cost for mixing index calculation and the actual mixing situation (Fig. 2(b)), we chose 1250 ($5 \times 5 \times 50$) subdomains for the entire domain of our 3D mixing system. Specifically, the 3D rotary drum was first divided into 50 layers with equal thickness of 3 mm each (equal to the particle diameter) in the axial direction. Then a square in which the circular cross section of the drum is inscribed has been defined, and a 5×5 equidistant square grid has been laid on this square. The detailed gridding manner is illustrated in Fig. 3.

Since the experimental images can only provide the location of particles on the front face, simulation results from only the first axial slice (the closest layer to the front wall) have been used for comparison. Both the used single

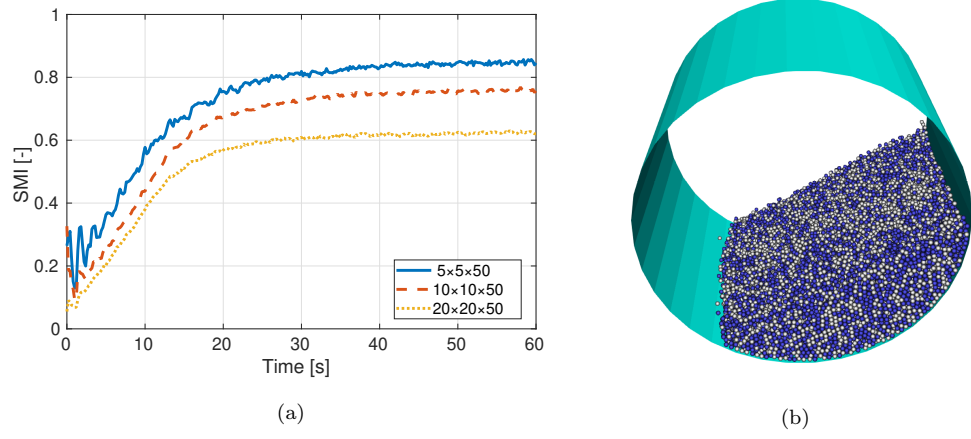


Figure 2: The effect of the number of subdomains on SMI: (a) evolution of indices with different numbers of subdomains (b) image of rotary drum at steady mixing state (32.5 s).

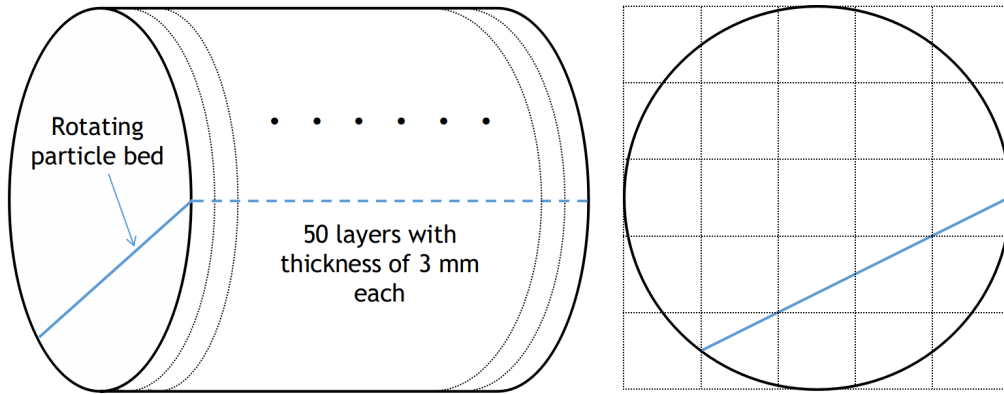


Figure 3: Gridding manner for SMI.

layer simulation data as well as the experimental data were further divided into 25 subdomains (5×5). Same gridding has also been used for the following 2D simulations.

Figure 4 shows the comparison of SMI evolution from experimental photos and the first layer of particles in the 3D benchmark DEM simulation. As can be seen, the mixing degrees at steady mixing state are quite close between experiment (0.88) and simulation (0.87). However, the mixing time in the experiment (22.7 s) is somewhat longer than in the simulation (18.3 s), which could be explained as follows: the revolution frequency of the rotary drum increases progressively from 0 to 15 rpm in the experiment due to inertia, whereas it reaches 15 rpm immediately in the simulation. In general though, the consistency of the two curves in Fig. 4 confirms that the 3D DEM simulation is able to replace the experiment with high reliability. Besides, the 3D DEM simulation can provide mixing information from the whole volume of the drum, which is hard to achieve in the experiment.

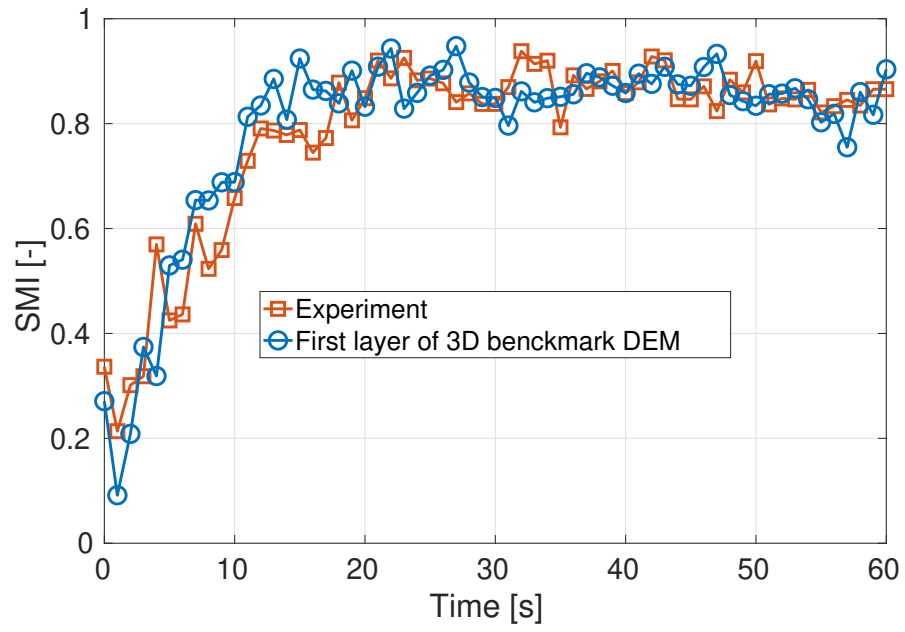


Figure 4: Comparison of experiment and the first layer of the 3D benchmark DEM simulation in terms of SMI.

3. Cross-correlation method

3.1. 3D DEM simulation vs. 2D DEM simulation

As pointed out in Section 1, 2D simulations are considered as a cheap surrogate model for expensive 3D simulations. Both 3D and 2D simulations were performed on a commercial computer with AMD Ryzen Threadripper 2950X processor (16 cores/32 threads, 3.5 GHz) and 32 GB RAM. On the one hand, for the 3D DEM simulations, 37600 particles in total were inserted into the simulation domain, and six hours of real time was spent for conducting 60 seconds of mixing process. On the other hand, 16 cases of 2D simulation of 60 seconds mixing process with 760 particles in parallel cost half an hour. The 2D simulations were conducted with same simulation parameters as in Table 1. In the 2D simulations, the particles are still spheres instead of discs, only the velocities and forces in the axial direction are zeroed out in every time step. There are no front and rear walls in the 2D simulations. Due to the lack of friction from these two surfaces, the motion regime in 2D is closer to slumping instead of rolling, which means that the particle bed is elevated and slides down periodically during the rotation of the drum, especially at low revolution frequency. It should be noted that, keeping other simulation settings identical but only changing the number of allocated CPU cores has certain influence on the final simulation results. Table 2 shows that, when the number of CPU cores changes, the 3D simulation results are quite stable. But the core number has a relatively large influence on the mixing time of 2D simulations. Hence, to increase the reliability of 2D simulation, every 2D case was, for otherwise same conditions, repeated three times with 1 core, 4 cores and 16 cores, respectively, as one set. The mixing time and mixing

degree at steady mixing state of the 2D simulations are the averaged values from these three runs. However, all the corresponding 3D simulations were only run once under 16 cores for saving simulation time. In summary, the time cost for running a set of 2D simulation is 1/64 of the time cost of one corresponding 3D simulation.

Table 2: The influence of allocated CPU cores on 3D and 2D DEM simulation.

	Core number [-]	Mixing time [s]	SMI at steady mixing state [-]
3D simulation	1	31.5	0.86
	4	31.3	0.85
	16	32.5	0.85
2D simulation	1	33.9	0.89
	4	38.1	0.88
	16	37.1	0.89

3.2. Development of cross-correlation

The 2D simulation seems to be able to well reflect mixing in that direction of the 3D rotary drum, in which mixing is by far more intense, namely in transverse direction. However, and expectedly, mixedness, including mixing time and mixing degree at steady state, is clearly not identical between 2D and 3D simulation. So the new idea is to develop a cross-correlation to bridge the differences in mixedness between the 2D (small, surrogate) and the 3D (large, target) particle system. Similar to conventional machine learning approaches, all the simulation data were first separated into two main parts:

learning set and prediction set, where the learning set is used for developing the cross-correlation and the prediction set is used for testing the predictive ability of the developed cross-correlation.

In the learning set, we chose the revolution frequency of the rotary drum as the unique input parameter, because the influence of revolution frequency on the mixing time is quite clear: higher revolution frequency leads to faster mixing, and this conclusion has been widely investigated in the literature (Van Puyvelde et al. (1999); Kwapinska et al. (2006); Liu et al. (2017)). Seventeen sets of 3D and 2D simulations were conducted. Compared to the benchmark 3D DEM simulation, only the revolution frequency was changed from 10 to 50 rpm with an increment of 2.5 rpm. The simulation results are plotted in Fig. 5.

For both 3D and 2D simulations, as can be seen in Fig. 5(a) and Fig. 5(b), the SMI at steady mixing state only fluctuates within a small range, indicating that the revolution frequency of the rotary drum has no significant influence on this index. Consequently, and for this reason, the connection between 3D and 2D simulations would be quite trivial regarding the mixing degree at steady state. Therefore, we have refrained from a respective cross-correlation, concentrating on the behavior of mixing time. As Fig. 5(c) and Fig. 5(d) illustrate, latter depends strongly on the revolution frequency for both the 3D and 2D simulations. Expectedly, the mixing time decreases with the increase in revolution frequency. However, the decreasing trend gradually slows down.

In both cases, mixing time can be correlated with the revolution frequency

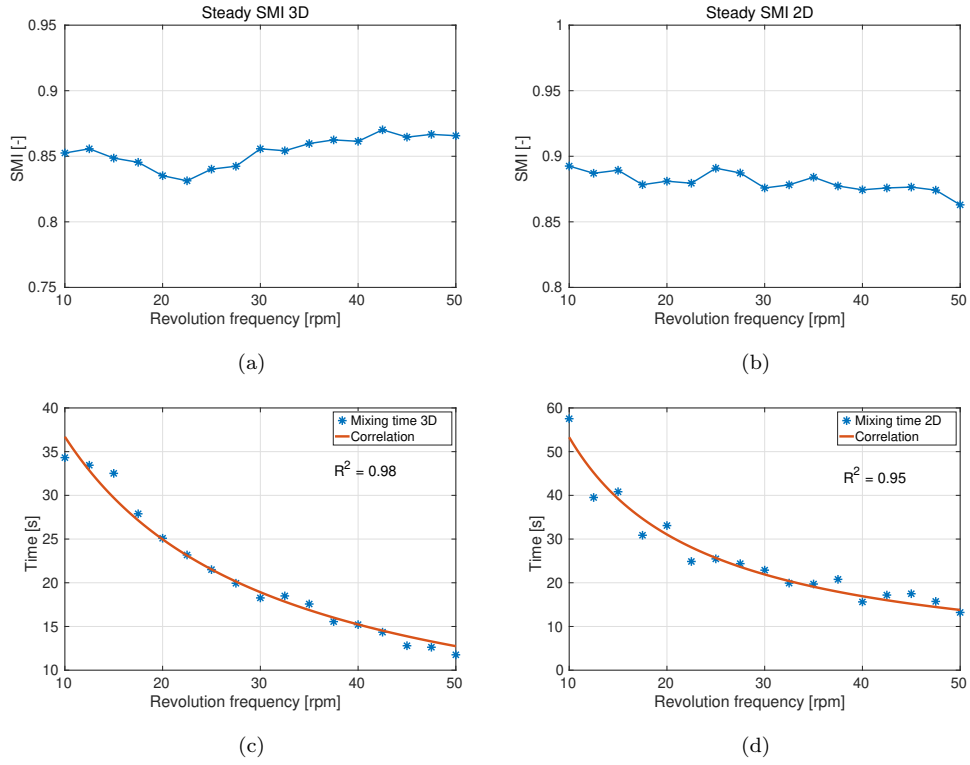


Figure 5: Relationships between mixedness and revolution frequency of rotary drum: (a) mixing degrees (SMI) at steady mixing state from 3D DEM simulations (b) mixing degrees (SMI) at steady mixing state from 2D DEM simulations (c) mixing times from 3D DEM simulations (d) mixing times from 2D DEM simulations.

f by a rational function, namely:

$$t_{3D} = \frac{781.1}{11.28 + f} \left(R^2 = 0.98 \right), \quad (6)$$

for the large 3D particle system, and

$$t_{2D} = \frac{743.8}{3.957 + f} \left(R^2 = 0.95 \right). \quad (7)$$

for its small 2D surrogate. Here, f is in rpm, times are in seconds. The values of the coefficient of determination (R^2) reflect a high consistency between the correlations and the original simulation data. Combining Eqs. (6) and (7), the mixing times of the 2D simulations and those of the 3D simulations (in seconds) can be cross-correlated to:

$$t_{3D} = \frac{781.1 t_{2D}}{743.8 + 7.323 t_{2D}}. \quad (8)$$

This is the final result of learning set considerations, namely the cross-correlation, which connects the mixing kinetics of the surrogate particle system with full-scale mixing kinetics, enabling to calculate the mixing time in 3D (t_{3D}) when the mixing time in 2D (t_{2D}) is known from respective, low cost simulations.

3.3. Test of predictive ability

3.3.1. Within the parametric range of learning set - revolution frequency

The developed cross-correlation along with the 2D simulation results constitute our tool box for predicting the mixing time of the large particle system considered (3D simulation). The test of predictive ability of the cross-correlation is firstly performed within the parametric range of the learning

set, which means for variation of the same parameter, the revolution frequency of the rotary drum, as in the learning set, and within the same range of values. This is done in two stages.

In the first stage, mixing times from the 3D simulations are compared to predictions of the cross-correlation at exactly the same revolution frequencies as in the learning set. Results of this comparison are presented in Table 3. As can be seen, the maximum prediction error is 4.3 s and the overall mean absolute error is 1.7 s. In addition, the cross-correlation seems to perform better under high revolution frequencies. A possible reason is that, under low revolution frequency, the motion regimes in 2D and 3D are different, namely rather slumping in 2D but rolling in 3D. Nevertheless, the coefficient of determination R^2 of these 17 predictions for the learning set is 0.92, which illustrates the high accuracy of the developed cross-correlation. In the second stage, comparisons are undertaken for three additional revolution frequencies that did not belong to the learning set. Results can be seen in Table 4, and are equally satisfactory.

3.3.2. Beyond the parametric range of learning set - density ratio

In the learning set, revolution frequency was the only parameter considered for developing the cross-correlation. Contrary, density of the two kinds of mixed particles was kept identical, in other words, their density ratio was always one. However, different particle densities are usually involved when mixing different ingredients in industrial processes. And, density ratio is also an important parameter that has a strong impact on the particle mixing behavior. When the density ratio is larger than one, segregation phenomena would appear (Pereira et al. (2011)). Denser particles tend to gather in the

Table 3: Mixing times at different drum revolution frequencies from 3D simulations and cross-correlation predictions.

Revolution frequency [rpm]	Mixing time from 3D simulation [s]	Mixing time from cross-correlation [s]	Error of prediction [s]
10	34.3	38.6	4.3
12.5	33.4	29.9	-3.5
15	32.5	30.6	-1.9
17.5	27.9	24.9	-3.0
20	25.1	26.2	1.1
22.5	23.2	21.0	-2.2
25	21.5	21.4	-0.1
27.5	19.9	20.6	0.7
30	18.3	19.6	1.3
32.5	18.5	17.5	-1.0
35	17.6	17.3	-0.3
37.5	15.6	18.1	2.5
40	15.2	14.2	-1.0
42.5	14.3	15.4	1.1
45	12.8	15.7	2.9
47.5	12.6	14.3	1.7
50	11.8	12.3	0.5

Table 4: Test of prediction with three additional simulations.

Revolution frequency [rpm]	Mixing time from 3D simulation [s]	Mixing time from cross-correlation [s]	Error of prediction [s]
16	30.3	33.1	2.8
32	18.1	21.1	3.0
48	12.1	13.0	0.9

central axis of the bed and form a heavy core with lighter particles surrounding. Figure 6 shows the mixing patterns in the 3D rotary drum when density segregation occurs (as visible from the outside, for a cross-sectional image see Fig. 8(a)). Conducted 3D simulations show that, the larger the density ratio is, the more obvious becomes the segregation phenomenon. The mechanism of density segregation is not completely clear, but this kind of segregation is generally attributed to buoyancy (Arntz et al. (2014)).

We would like to test whether our developed cross-correlation is still applicable in this density segregation situation, or not. For this purpose, 3D and 2D simulations with density ratio from 1.25 to 2 at different revolution frequencies were performed. To better represent the changing trend of mixedness, five data points with density ratio of one from the previous learning set are also plotted together with the new results in Fig. 7. The new results correspond to 20 data sets with unequal particle density.

Figures 7(a) and 7(b) show that, for both 3D and 2D simulations, the mixing degree at steady state decreases with increasing density ratio when the revolution frequency is constant. When the density ratio is larger than one, higher revolution frequency would promote the mixing process and finally

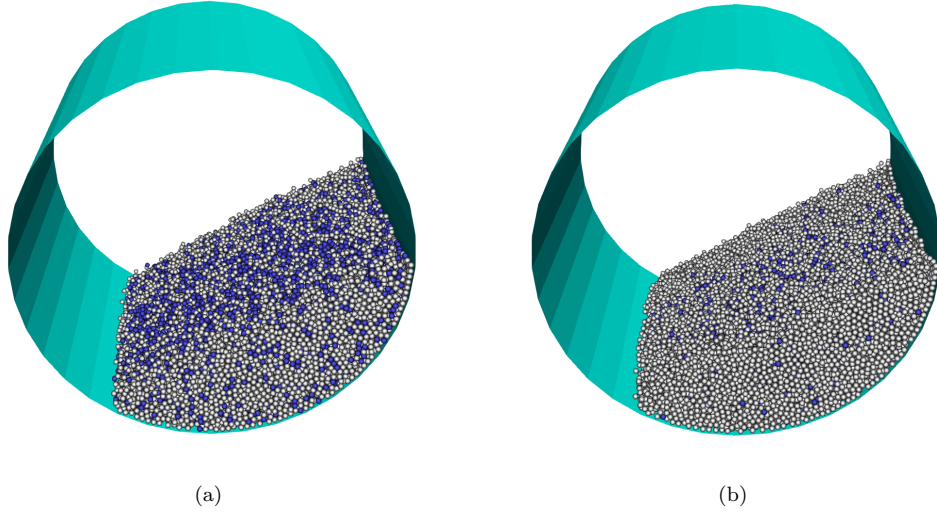


Figure 6: Snap-shots of mixture from 3D DEM simulations with different density ratio: (a) density ratio = 1.25 at steady mixing state (b) density ratio = 2 at steady mixing state (blue particles are heavier than white particles).

lead to a higher mixing degree in 3D. However, this effect is not observed in 2D. Figures 7(c) and 7(d) indicate quite similar changing trends of mixing time for 3D and 2D simulations. Mixing time generally decreases by increasing either the density ratio or the revolution frequency.

As before, mixing times obtained from the 3D simulation and mixing times predicted from the cross-correlation have, then, been compared. This comparison is summarized in Table 5. We can find that the largest prediction errors are still concentrated at low revolution frequency, similar to the previous predictions for the learning set. Here, the maximum prediction error is with 9.0 s quite large, but the overall mean absolute error is with 3.7 s significantly smaller. The coefficient of determination of these 20 new predictions is with $R^2 = 0.86$ satisfactory. This proves that the cross-correlation is

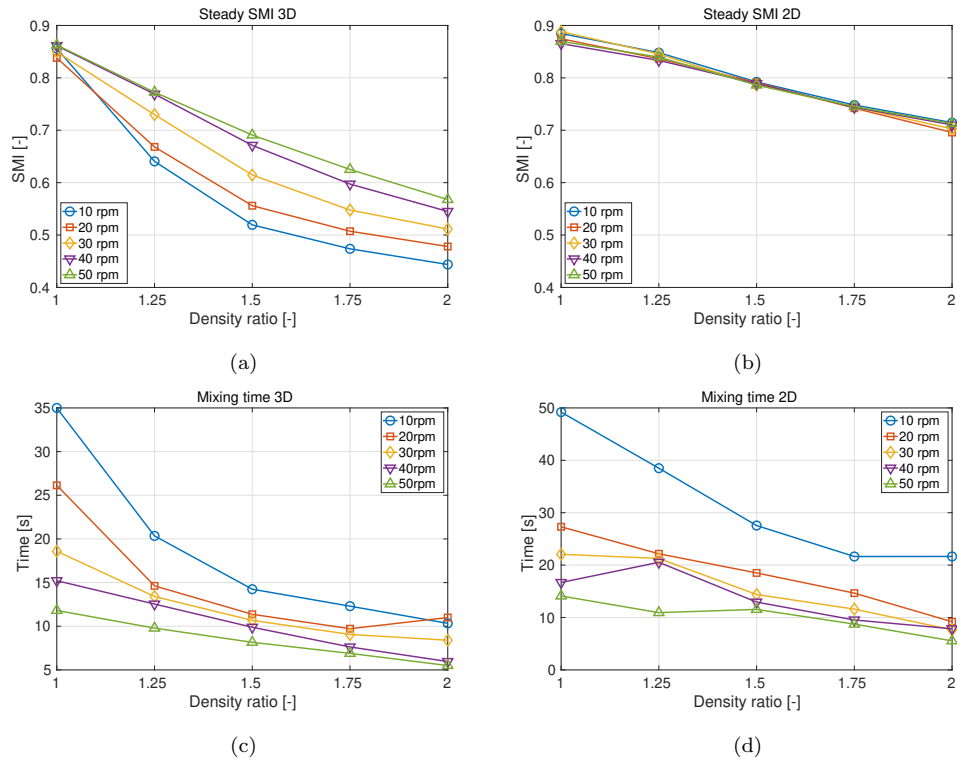


Figure 7: Relationships between mixedness and density ratio at different revolution frequencies: (a) mixing degrees (SMI) at steady mixing state from 3D DEM simulations (b) mixing degrees (SMI) at steady mixing state from 2D DEM simulations (c) mixing times from 3D DEM simulations (d) mixing times from 2D DEM simulations.

still acceptably accurate and usable for situations with density ratio different than unity.

These satisfactory results are not a coincidence, and we can find the reason behind them by observing the mixing patterns of 3D and 2D drums (Fig. 8). When both simulations have reached a steady mixing state, the cross section at mid length of the drum from the 3D simulation is contrasted in Fig. 8 with the cross section of the 2D simulation. It can be observed that these two cross sectional images are quite similar, the heavier blue particles being concentrated in the middle while the lighter white particles are surrounding them in form of an outer ring. Although this phenomenon is more clear in 3D, it is also present in 2D. Put together, Figures 6(b) and 8(a) provide a complete picture of density segregated mixture in 3D drum. They show that the density segregation is really stable in the transverse direction. Contrary, there is, except for the particles close to the front and rear wall, which almost all are lighter white particles, almost no change in particle concentration in the axial direction. Obviously, the overall mixing situation in the 3D drum is dominated by the transverse mixing. This is why the connection of mixing time between 3D and 2D simulations is still existent when with the density ratio a new parameter is taken into account.

3.3.3. Beyond the parametric range of learning set - particle size ratio

Besides density difference, binary mixtures with different particle sizes are also of practical interest and could also cause segregation. Differently from the density segregation, size segregation though occur in both transverse direction and axial direction (Ding et al. (2002), Kuo et al. (2005)). In the transverse plane, small particles tend to concentrate in the core re-

Table 5: Mixing times with density ratio larger than one at different revolution frequencies from 3D simulations and cross-correlation predictions.

f [rpm]	Density ratio [-]	Mixing time from 3D simulation [s]	Mixing time from cross-correlation [s]	Error of prediction [s]
10	1.25	20.3	29.3	9.0
	1.5	14.2	22.8	8.5
	1.75	12.3	18.7	6.4
	2	10.3	18.7	8.4
20	1.25	14.6	19.1	4.5
	1.5	11.4	16.4	5.1
	1.75	9.7	13.4	3.7
	2	11.0	8.9	-2.1
30	1.25	13.4	18.5	5.1
	1.5	10.7	13.2	2.6
	1.75	9.1	10.9	1.9
	2	8.4	7.5	-0.9
40	1.25	12.5	17.9	5.4
	1.5	9.9	12.1	2.2
	1.75	7.6	9.1	1.5
	2	6.0	7.7	1.7
50	1.25	9.8	10.4	0.6
	1.5	8.2	10.9	2.7
	1.75	6.9	8.5	1.6
	2	5.5	5.5	0.0

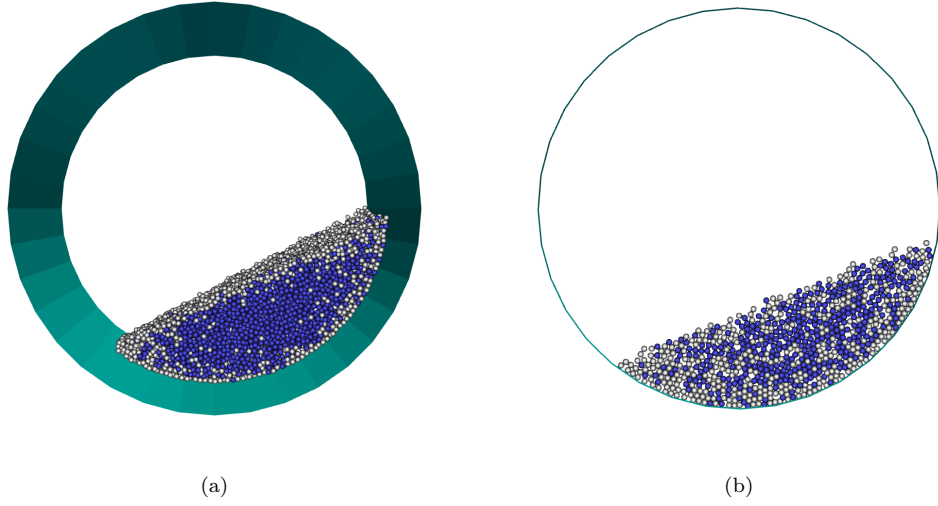


Figure 8: Mixing patterns of 3D and 2D simulation when density segregation occurs (density ratio = 2): (a) cross section at the middle of drum's length in the 3D simulation (b) cross section of the 2D simulation.

gion, whereas large particles tend to occupy the shell region, which could be attributed to dynamic sieving (Arntz et al. (2014)). In the axial direction, a pattern of alternating bands would form. Small particles are entrapped in the body of the particle bed, whereas large particles accumulate at the front and rear walls, which may be caused by the end wall effect (Cui et al. (2014)).

In order to test the predictive ability of the cross-correlation when the size segregation appears, 3D and 2D simulations with particle size ratio equal to two were performed, where the large particles had a diameter of 6 mm and the small particles were 3 mm in diameter. The filling degree remains at 20% and the revolution frequency is 10 rpm. Figure 9 is a comparison of mixing patterns between 3D and 2D simulation when size segregation

occurs. Clearly, the 2D simulation cannot reflect the axial segregation due to the particle movement, which in this case is restricted in the transverse plane. Next we analyzed the SMI curves of 3D and 2D simulations, as shown in Fig. 10. It is interesting to see that when size segregation occurs, the SMI curve of the 3D simulation is totally different than before. After a short period of elevation at the beginning, as the alternating bands form in the axial direction, the mixing degree gradually decreases and finally reaches a steady state. However, the trend of the mixing degree from the 2D simulation is still similar to the trend without size segregation - with mixing degree that increases, apart from fluctuations, monotonically in time. The value of mixing time that the cross-correlation predicts for the large particle system would be 20.4 s in this case, but this value has not much in common with the behavior of the large system that the 3D DEM simulation shows. So, the cross-correlation is not appropriate in cases that feature particle size ratio significantly different from unity. This is mainly because the 2D simulation cannot reflect the mixing process in the axial direction.

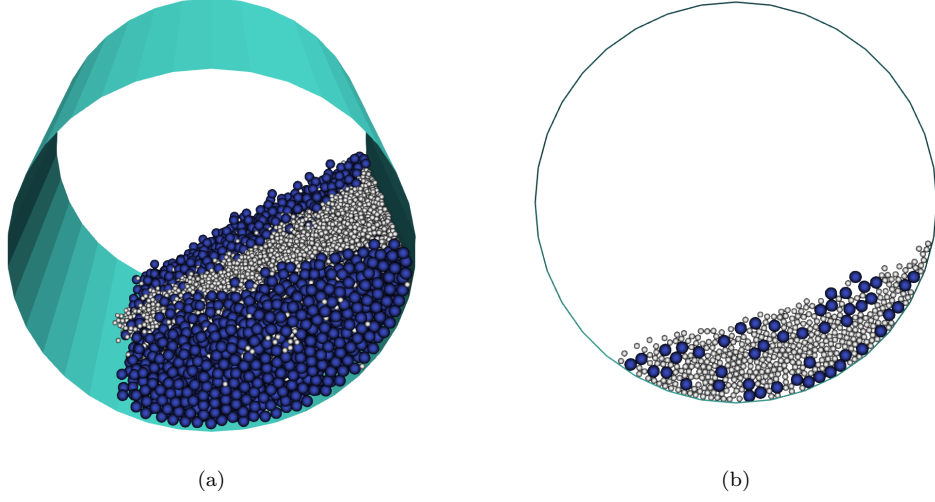


Figure 9: Mixing patterns of 3D and 2D simulation when size segregation occurs (particle size ratio = 2): (a) snapshot of 3D simulation at 60 s (b) snapshot of 2D simulation at 60 s.

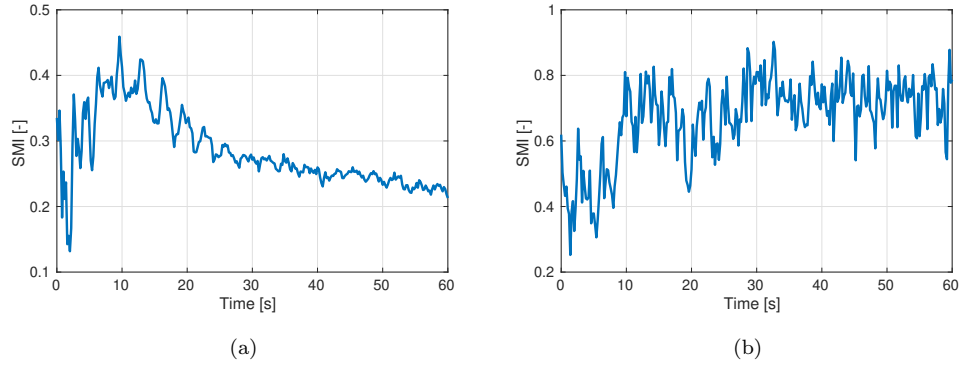


Figure 10: SMI evolution of 3D and 3D simulation when size segregation occurs (particle size ratio = 2): (a) SMI curve of 3D simulation (b) SMI curve of 2D simulation.

4. Discussion

Recapitulating, the presented results have shown that the idea of approximating the mixing behavior of a large particle system by the mixing behavior of a much smaller surrogate system and a scale-bridging cross-correlation is, in principle, feasible. Taking DEM simulations of a 3D rotary drum as the large system, and similar simulations in 2D as the small system, large system behavior could be accurately predicted by means of computationally cheap small system simulations and the previously derived cross-correlation. This has been the case when the learning data set of the method was used, but also for other conditions within the parametric range of the learning set. Positively, perhaps unexpectedly positive, the new approach and the cross-correlation have been found to hold even outside of the parametric range of the learning set, for the mixing of particulate species of different density. Satisfactory predictive performance in mentioned cases is summarized by means of a parity plot in Fig. 11. Though, there are obviously also limits to the applicability of the concept, which have been reached and illustrated by considering particulate species of different size.

In this context, comparison is interesting of the new concept with black-box machine learning methods, such as artificial neural networks ([Kumar et al. \(2018\)](#), [Kalathingal et al. \(2020\)](#)) or gradient boosting models ([Qi et al. \(2018\)](#)). Methods of this kind are becoming increasingly popular for predicting the behavior of big and complex systems of various kinds in industry, and they can indeed provide excellent accuracy after training by massive data. Compared to these black-box models, the cross-correlation method is quite simple and can be derived from very limited sets of data, but still pro-

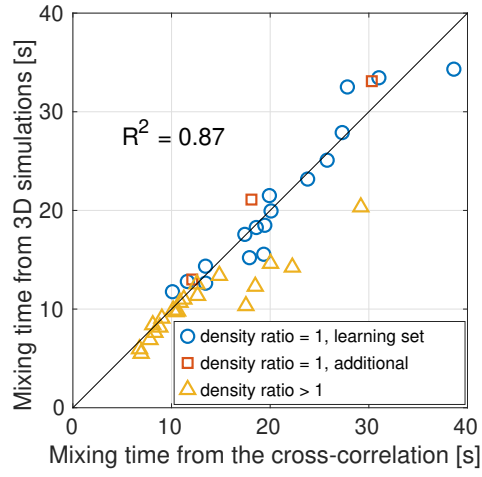


Figure 11: Parity plot of mixing time from 3D simulation and the cross-correlation prediction (including 17 points that correspond to the learning set; 3 points of additional simulations that do not belong to the learning set but within its parametric range; 20 points that correspond to unequal density, outside of the parametric range of the learning set).

viding satisfactory predictive ability. Moreover, the cross-correlation method has two notable merits, interpretability and expandability. Interpretability stems from the physical foundation of the cross-correlation. For example, when particles are mixed in a rotary drum, the transverse mixing is much more intense than the axial mixing. The 2D simulation can well reflect particle mixing in the transverse plane and the trend of its mixing time is similar to that of the 3D simulation when revolution frequency changes, which allows the cross-correlation to be successfully developed. The expandability of the cross-correlation is embodied in its already discussed potential to make predictions beyond the parametric range of its learning set, where conventional machine learning models are usually unable to do this. With the here discussed aim of reconstructing the mixing behavior of particles in large rotary drums, the expandability is limited when the axial mixing effect becomes significant. However, the cross-correlation method might still successfully perform outside of the parametric range of its learning set, capturing the effects of further influential parameters such as filling degree, cohesive particles or additional blades. Whereas this, as well application to different kinds of mixers at both, the large and the small scale, remain to be seen, even the limited scope of the present work indicates new possibilities by the proposed concept for fast and cheap predictions of the operation of industrial mixers in the future.

5. Conclusion

A new concept for the prediction of the mixing time of particles in large mixers from small system mixing behavior with the help of a scale-spanning

cross-correlation has been prototypically explored. In the worked out example, the large system is a 3D rotary drum, whereas the small system is a 2D rotary drum. Data at both scales are generated by means of DEM simulations. Main findings are:

- The cross-correlation along with the 2D simulation is a cheap surrogate for the expensive 3D simulation, 64 times cheaper in terms of computational time cost.
- The development of the cross-correlation is based on 17 sets of simulation data (learning set) gained with variation of the revolution frequency. Predictions of mixing time within the parametric range of the learning set are very satisfactory ($R^2 = 0.92$).
- Predictions of mixing time beyond the parametric range of learning set were also tested. Good results ($R^2 = 0.86$) are obtained for the mixing time of particles that differ in density.
- Contrary, the cross-correlation fails when particulate species of different size are considered. This is due to the fact that segregation caused by particle size difference in the axial direction cannot be neglected. So we speculate that as long as the newly added parameter has no significant impact on the axial mixing, the cross-correlation could work well.

Further generalization of the concept, as well as further investigation of its applicability range and limits should be addressed in future work.

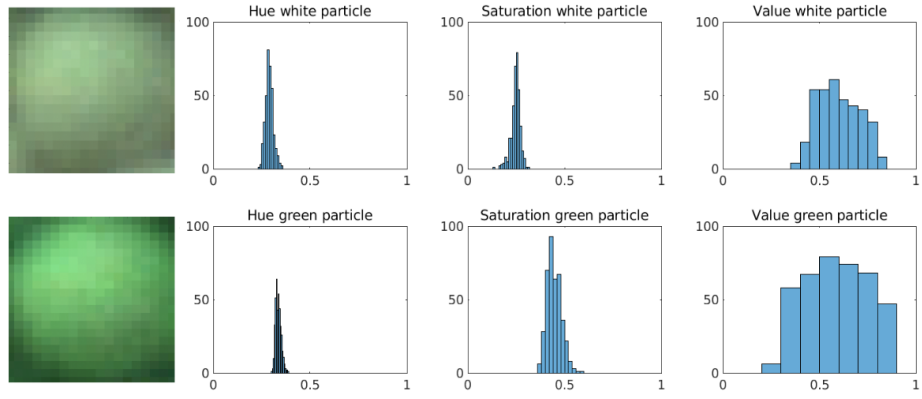
Acknowledgments

The authors would like to gratefully acknowledge the funding of this work by the German Federal Ministry of Science and Education (BMBF) within the project ADMIX (03INT609BA).

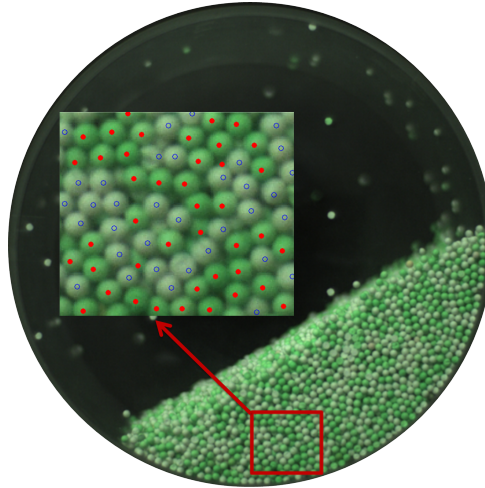
Appendix A. Image processing

In order to get the center positions of each white or green particle in the images, the Hue-Saturation-Value (HSV) color model was used, which is easier and more accurate compared with the Red-Green-Blue (RGB) color model (Olafe et al. (2013)). The image processing steps are briefly introduced as follows:

1. Importing the experimental images into MATLAB, and then transforming the RGB information of every pixel into the HSV space by using the `rgb2hsv` function.
2. Selecting a representative white and a representative green particle from the image manually. Based on the different range of their HSV values, the white part and the green part in the image can then be identified (Fig. A.1(a)), for more details see Jiang et al. (2018).
3. With the help of a normalized 2D cross-correlation algorithm and with known value of ideal particle diameter, the center coordinates of white and green particles were obtained as shown in Fig. A.1(b), where the red dots represent the centers of the green particles, and the hollow blue points represent the centers of white particles.



(a)



(b)

Figure A.1: Image processing: (a) histograms of HSV channels for representative white and green particles (b) particle center determination.

Appendix B. DEM governing equations

Except of gravity, the force between two particles in contact with each other was taken into account, decomposed in normal direction n and tangential direction t :

$$\mathbf{F} = \mathbf{F}_n + \mathbf{F}_t. \quad (\text{B.1})$$

In the soft-sphere contact model, overlap is assumed to be possible between two contacting particles. The normal force F_n , on the one hand, can be expressed as:

$$F_n = k_n \delta_{n_{i,j}} - \gamma_n v_{n_{ij}}. \quad (\text{B.2})$$

As can be seen, the normal force can be further divided into two terms: the spring force and the damping force. The spring force is the product of the elastic constant for normal contact (k_n) and the normal overlap distance of particle i and particle j ($\delta_{n_{i,j}}$), whereas the damping force is the product of the viscoelastic damping constant for normal contact (γ_n) and the normal component of the relative velocity of the two particles ($v_{n_{ij}}$). On the other hand, the tangential force also consists of two terms: the shear force and the damping force,

$$F_t = k_t \delta_{t_{i,j}} - \gamma_t v_{t_{ij}}. \quad (\text{B.3})$$

Here, k_t is the elastic constant for tangential contact, $\delta_{t_{i,j}}$ is tangential overlap, γ_t is the viscoelastic damping constant for tangential contact and $v_{t_{ij}}$ represents the tangential component of the relative velocity of the two particles. In addition, the tangential overlap is truncated to fulfill Coulomb's law of friction:

$$|F_t| \leq \mu |F_n|, \quad (\text{B.4})$$

where μ is the friction coefficient between two particles. Hertz contact model with tangential loading history condition was chosen to describe particle interaction (Di Renzo and Di Maio (2004); Navarro and de Souza Braun (2013)). In this model, k_n , k_t , γ_n and γ_t can be calculated through material properties as follows:

$$k_n = \frac{4}{3}E_{eq}\sqrt{r_{eq}\delta_{n_{i,j}}}, \quad (\text{B.5})$$

$$k_t = 8G_{eq}\sqrt{r_{eq}\delta_{n_{i,j}}}, \quad (\text{B.6})$$

$$\gamma_n = \sqrt{5}\alpha\sqrt{k_n m_{eq}}, \quad (\text{B.7})$$

$$\gamma_t = \sqrt{\frac{10}{3}}\alpha\sqrt{k_t m_{eq}}, \quad (\text{B.8})$$

$$\alpha = -\frac{\ln e}{\sqrt{\ln^2 e + \pi^2}}, \quad (\text{B.9})$$

where e is the coefficient of restitution between two particles and α is the dimensionless damping coefficient. E_{eq} , r_{eq} , G_{eq} and m_{eq} are the four equivalent properties, which respectively represent Young's modulus, radius, shear modulus and mass of two contacting particles. These equivalent properties can be expressed as follows:

$$\frac{1}{E_{eq}} = \frac{1 - \nu_i^2}{E_i} + \frac{1 - \nu_j^2}{E_j}, \quad (\text{B.10})$$

$$\frac{1}{r_{eq}} = \frac{1}{r_i} + \frac{1}{r_j}, \quad (\text{B.11})$$

$$\frac{1}{G_{eq}} = \frac{2 - \nu_i}{G_i} + \frac{2 - \nu_j}{G_j}, \quad (\text{B.12})$$

$$G_i = \frac{E_i}{2(1 + \nu_i)}, \quad (\text{B.13})$$

$$\frac{1}{m_{eq}} = \frac{1}{m_i} + \frac{1}{m_j}, \quad (\text{B.14})$$

where ν_i and ν_j are the Poisson's ratios of the two particles. As for the torque, which controls the particle rotational motion, it can also be separated into two parts: one caused by the contact forces between particles and another one attributed to the rolling friction. The constant directional torque model (Ai et al. (2011)) was selected to describe the torque contribution from the rolling friction:

$$\mathbf{M}_{i,j} = -k_r k_n \delta_{n_{i,j}} \frac{\boldsymbol{\omega}_{rel}}{|\boldsymbol{\omega}_{rel}|} r_{eq}, \quad (\text{B.15})$$

where k_r is the rolling coefficient, $\boldsymbol{\omega}_{rel}$ is the relative angular velocity between two contact particles.

Appendix C. DEM parameters determination

Simulations from Yamamoto et al. (2016) indicated that Young's modulus, Poisson's ratio and coefficient of restitution are not of essential importance for particles motion. Alizadeh et al. (2014) also proved that DEM simulation results are still physically sound when unrealistic particle properties are used. Young's modulus is related to the setting of time step. A lower value of Young's modulus was, therefore, used in order to save the simulation time. Poisson's ratio of γ -alumina particles was taken from Auerkari (1996)

and the coefficient of restitution from measurements conducted by [Mueller et al. \(2011\)](#) with an electromagnetic induction canon.

Friction can strongly affect the particle mixing behavior. Smooth surface of γ -alumina particle makes the rolling resistance really weak in the mixing system, so the rolling coefficient was set as 0.01. The determination of friction coefficient is based on the comparison of the dynamic angle of repose between experiment and DEM simulation. When the particle bed in the drum rotates in rolling motion, which is the most recommended motion regime for the mixing process, a so-called active layer of particle flow is formed on the bed surface ([Mellmann \(2001\)](#)). The angle between inclined bed surface and the horizontal is defined as the dynamic angle of repose (α). Under rolling motion, a greater friction coefficient would lead to a higher dynamic angle of repose.

In order to find the DEM simulation that can represent the experiment, several trial simulations with different friction coefficients were conducted, and then the dynamic angles of repose from experiment and simulations were compared. In the experiment, after image processing, the positions of particles at the transparent front face of the drum were stored. In the simulations, the coordinates of every single particle in the drum can be obtained, but only the first slice at the front face with thickness of one particle diameter (3 mm) was considered for angle recording. A Cartesian coordinate system was placed at the center of the cross section of the anticlockwise rotating drum and the dynamic angle of repose was derived from two specific particles: the particle with smallest coordinate value in x direction, and the particle with biggest coordinate value in y direction, as shown in Fig. [C.1](#).

In the experiment, some particles which stuck electrostatically on the inner wall of the drum were excluded from angle determination. The final dynamic angle of repose is a time averaged value from 2 s to 10 s, during which the angle fluctuation is relatively small. Figure C.2 gives an example for the evolution of repose angle from a DEM simulation over time.

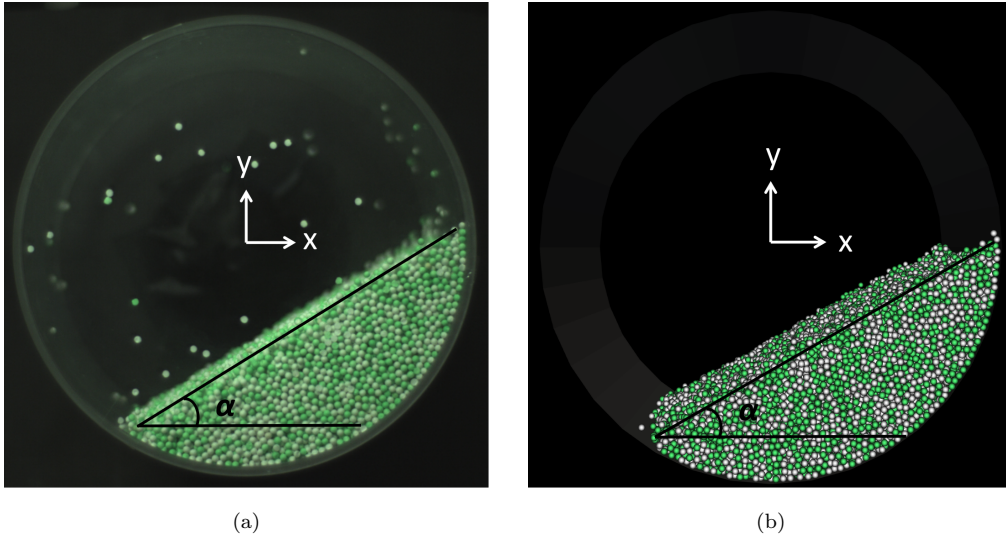


Figure C.1: Dynamic angle of repose α in: (a) experiment (b) 3D DEM simulation (filling degree 20%, revolution frequency 15 rpm).

The results of trial simulations with different friction coefficients are listed in Table C.1. The friction coefficient was finally determined as 0.8 for all subsequent simulations, where the dynamic angle of repose from the simulation (30.4°) is closest to the experimental value (30.6°).

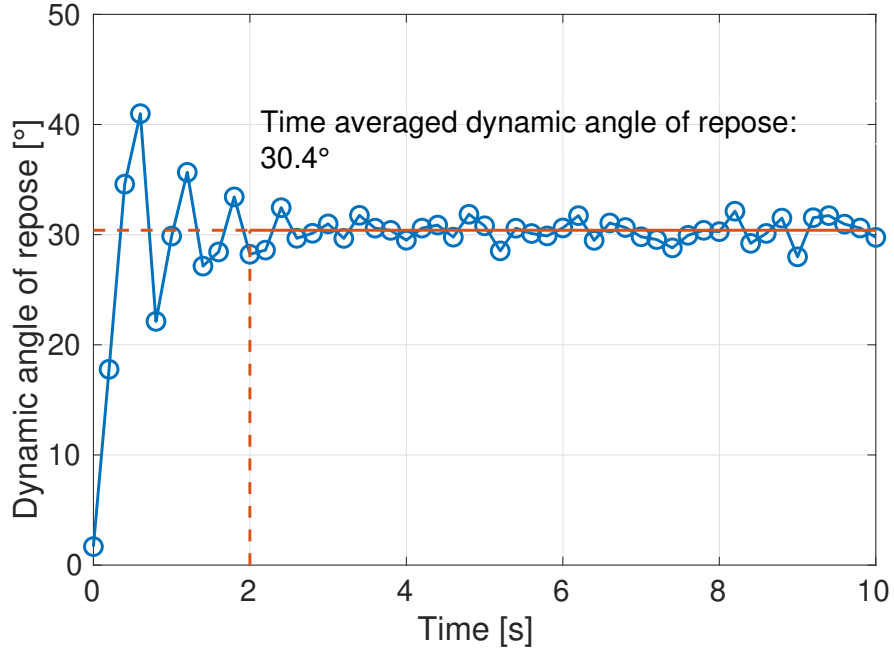


Figure C.2: Evolution and average of dynamic angle of repose.

Table C.1: Average dynamic angle of repose under different conditions.

	Friction coefficient	Dynamic angle of repose [°]
Experiment	-	30.6
	0.5	26.7
	0.6	28.8
DEM simulation	0.7	29.8
	0.8	30.4
	0.9	30.9

Nomenclature

Symbols

a, b, c	Parameters in fitted curves	-
e	Coefficient of restitution	-
E	Young's modulus	Pa
f	Revolution frequency	rpm
f_k	Participation factor of particle type k	-
F, \mathbf{F}	Force	N
G	Shear modulus	Pa
k	Elastic constant	kg/s ²
k_r	Rolling coefficient	-
L	Lacey index	-
m	Mass	kg
\mathbf{M}	Torque from rolling friction	N · s
n	Number of subdomains	-
N	Number of particles	-
p	Number fraction of particles	-
P	Intermediate fraction in SMI	-
Q	Types of particles	-
r	Radius of particle	m
R	Density ratio	-
R^2	Coefficient of determination	-
s^2	Variance	-
S	mixing entropy index	-
t	Time	s

v	Velocity	m/s
x	Number fraction of particles	-
Greek symbols		
α	Damping coefficient	-
γ	Viscoelastic damping constant	kg/s
δ	Overlap distance of two particles	m
μ	Friction coefficient	-
ν	Poisson's ratio	-
ρ	Density	kg/m ³
ω	Angular velocity	rad/s
Subscripts		
0	Unmixed	
1, 2, 3	Different particle types	
2D	Two-dimensional	
3D	Three-dimensional	
10, 30, 50	Different revolution frequency	
<i>add</i>	Additional	
<i>bas</i>	Basic	
<i>cc</i>	Cross-correlation	
<i>eq</i>	Equivalent	
<i>i, j, k</i>	Indices	
<i>mix</i>	Mixing	
<i>n</i>	Normal direction	
<i>r</i>	Fully mixed	
<i>rel</i>	Relative	

<i>sim</i>	Simulation
<i>t</i>	Tangential direction
<i>tot</i>	Total
Abbreviations	
CFD	Computational fluid dynamics
DEM	Discrete element method
HSV	Hue-Saturation-Value
RGB	Red-Green-Blue
SMI	Subdomain-based mixing index

References

- Ai, J., J.-F. Chen, J. M. Rotter, and J. Y. Ooi (2011). Assessment of rolling resistance models in discrete element simulations. *Powder Technology* 206(3), 269–282.
- Alizadeh, E., F. Bertrand, and J. Chaouki (2014). Comparison of DEM results and Lagrangian experimental data for the flow and mixing of granules in a rotating drum. *AIChE Journal* 60(1), 60–75.
- Arntz, M., H. Beftink, W. K. den Otter, W. J. Briels, and R. Boom (2014). Segregation of granular particles by mass, radius, and density in a horizontal rotating drum. *AIChE Journal* 60(1), 50–59.
- Auerkari, P. (1996). *Mechanical and physical properties of engineering alumina ceramics*. Technical Research Centre of Finland Espoo.
- Bertrand, F., L.-A. Leclaire, and G. Levecque (2005). DEM-based models for the mixing of granular materials. *Chemical Engineering Science* 60(8-9), 2517–2531.
- Chen, H., Y. Xiao, Y. Liu, and Y. Shi (2017). Effect of young’s modulus on DEM results regarding transverse mixing of particles within a rotating drum. *Powder Technology* 318, 507–517.
- Cho, M., P. Dutta, and J. Shim (2017). A non-sampling mixing index for multicomponent mixtures. *Powder Technology* 319, 434–444.
- Cui, Z., Y. Zhao, Y. Chen, X. Liu, Z. Hua, C. Zhou, and J. Zheng (2014).

- Transition of axial segregation patterns in a long rotating drum. *Partic-
uology* 13, 128–133.
- Cundall, P. A. and O. D. Strack (1979). A discrete numerical model for
granular assemblies. *Geotechnique* 29(1), 47–65.
- Deen, N. G., G. Willem, G. Sander, and J. Kuipers (2010). Numerical analy-
sis of solids mixing in pressurized fluidized beds. *Industrial & Engineering
Chemistry Research* 49(11), 5246–5253.
- Di Renzo, A. and F. P. Di Maio (2004). Comparison of contact-force models
for the simulation of collisions in DEM-based granular flow codes. *Chemical
Engineering Science* 59(3), 525–541.
- Ding, Y., R. Forster, J. Seville, and D. Parker (2002). Segregation of granular
flow in the transverse plane of a rolling mode rotating drum. *International
Journal of Multiphase Flow* 28(4), 635–663.
- Finnie, G., N. P. Kruyt, M. Ye, C. Zeilstra, and J. Kuipers (2005). Longi-
tudinal and transverse mixing in rotary kilns: A discrete element method
approach. *Chemical Engineering Science* 60(15), 4083–4091.
- Govender, N., D. N. Wilke, C.-Y. Wu, R. Rajamani, J. Khinast, and B. J.
Glasser (2018). Large-scale GPU based DEM modeling of mixing using
irregularly shaped particles. *Advanced Powder Technology* 29(10), 2476–
2490.
- Hlosta, J., L. Jezerská, J. Rozbroj, D. Žurovec, J. Nečas, and J. Zegzulka
(2020). DEM investigation of the influence of particulate properties and

- operating conditions on the mixing process in rotary drums: Part 1—determination of the DEM parameters and calibration process. *Processes* 8(2), 222.
- Jiang, Z., T. Hagemeyer, A. Bück, and E. Tsotsas (2018). Color-PTV measurement and CFD-DEM simulation of the dynamics of poly-disperse particle systems in a pseudo-2D fluidized bed. *Chemical Engineering Science* 179, 115–132.
- Kalathingal, M. S. H., S. Basak, and J. Mitra (2020). Artificial neural network modeling and genetic algorithm optimization of process parameters in fluidized bed drying of green tea leaves. *Journal of Food Process Engineering* 43(1), e13128.
- Kloss, C., C. Goniva, A. Hager, S. Amberger, and S. Pirker (2012). Models, algorithms and validation for opensource DEM and CFD-DEM. *Progress in Computational Fluid Dynamics, an International Journal* 12(2-3), 140–152.
- Kumar, R., C. M. Patel, A. K. Jana, and S. R. Gopireddy (2018). Prediction of hopper discharge rate using combined discrete element method and artificial neural network. *Advanced Powder Technology* 29(11), 2822–2834.
- Kuo, H., R. Hsu, and Y. Hsiao (2005). Investigation of axial segregation in a rotating drum. *Powder technology* 153(3), 196–203.
- Kwapinska, M., G. Saage, and E. Tsotsas (2006). Mixing of particles in rotary drums: A comparison of discrete element simulations with exper-

- imental results and penetration models for thermal processes. *Powder Technology* 161(1), 69–78.
- Lacey, P. M. C. (1954). Developments in the theory of particle mixing. *Journal of Applied Chemistry* 4(5), 257–268.
- Liu, X., Z. Hu, W. Wu, J. Zhan, F. Herz, and E. Specht (2017). DEM study on the surface mixing and whole mixing of granular materials in rotary drums. *Powder Technology* 315, 438–444.
- Mellmann, J. (2001). The transverse motion of solids in rotating cylinders—forms of motion and transition behavior. *Powder Technology* 118(3), 251–270.
- Mueller, P., S. Antonyuk, M. Stasiak, J. Tomas, and S. Heinrich (2011). The normal and oblique impact of three types of wet granules. *Granular Matter* 13(4), 455–463.
- Navarro, H. A. and M. P. de Souza Braun (2013). Determination of the normal spring stiffness coefficient in the linear spring-dashpot contact model of discrete element method. *Powder Technology* 246, 707–722.
- Nityanand, N., B. Manley, and H. Henein (1986). An analysis of radial segregation for different sized spherical solids in rotary cylinders. *Metallurgical and Materials Transactions B* 17(2), 247–257.
- Olaofe, O., K. Buist, N. Deen, M. van der Hoef, and J. Kuipers (2013). Improved digital image analysis technique for the evaluation of segregation in pseudo-2D beds. *Powder Technology* 244, 61–74.

- Pereira, G. G., M. D. Sinnott, P. Cleary, K. Liffman, G. Metcalfe, and I. D. Šutalo (2011). Insights from simulations into mechanisms for density segregation of granular mixtures in rotating cylinders. *Granular Matter* 13(1), 53–74.
- Qi, C., Q. Chen, A. Fourie, J. Zhao, and Q. Zhang (2018). Pressure drop in pipe flow of cemented paste backfill: Experimental and modeling study. *Powder Technology* 333, 9–18.
- Schutyser, M., J. Padding, F. Weber, W. J. Briels, A. Rinzema, and R. Boom (2001). Discrete particle simulations predicting mixing behavior of solid substrate particles in a rotating drum fermenter. *Biotechnology and Bioengineering* 75(6), 666–675.
- Van Puyvelde, D., B. Young, M. Wilson, and S. Schmidt (1999). Experimental determination of transverse mixing kinetics in a rolling drum by image analysis. *Powder Technology* 106(3), 183–191.
- Xu, Y., C. Xu, Z. Zhou, J. Du, and D. Hu (2010). 2D DEM simulation of particle mixing in rotating drum: A parametric study. *Particuology* 8(2), 141–149.
- Yamamoto, M., S. Ishihara, and J. Kano (2016). Evaluation of particle density effect for mixing behavior in a rotating drum mixer by DEM simulation. *Advanced Powder Technology* 27(3), 864–870.
- Yan, B. and R. Regueiro (2018). Large-scale dynamic and static simulations of complex-shaped granular materials using parallel three-dimensional dis-

crete element method DEM on DoD supercomputers. *Engineering Computations* 35(2), 1049–1084.

Zhang, L., Z. Jiang, F. Weigler, F. Herz, J. Mellmann, and E. Tsotsas (2020). PTV measurement and DEM simulation of the particle motion in a flighted rotating drum. *Powder Technology* 363, 23–37.

Chapter 3

Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model

The cross-correlation method of Chapter 2 is limited in that it cannot predict the mixing degree at steady mixing state and only accounts for two influencing factors: revolution frequency and density ratio. To overcome these limitations and expand the applicability, a machine learning model was introduced as an alternative prediction approach. This model incorporates size ratio and drum length as additional input parameters and extends the prediction scope to include the mixing degree at steady mixing state.

A paper with the same title as this chapter has already been published:

Wencong Wu, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model.” *Powder Technology* 434 (2024): 119365.

As this paper is not open access, the attached version is the preprint manuscript containing the same content as the published version. For highlights and the graphical abstract, please visit:

<https://doi.org/10.1016/j.powtec.2024.119365>.

As the first author, my **individual contributions** include: designing and performing all DEM simulations, analyzing the simulation data, developing the machine learning model—PSO-SVR for mixing prediction, and writing the original draft with all figures and tables.

Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model

W. Wu*, K. Chen, E. Tsotsas

*Thermal Process Engineering, Otto von Guericke University, Universitätsplatz 2, 39106
Magdeburg, Germany*

Abstract

Particle mixing in rotary drums is of significant industrial importance, but very complex. Detailed simulation can be achieved using the discrete element method (DEM), but the computational time cost is enormous. Therefore, we combined machine learning with DEM simulations and established a DEM data-driven particle swarm optimization and support vector regression (PSO-SVR) model to predict mixing time and mixing degree at steady mixing state for binary sphere mixtures in horizontal rotary drums with four independent variables: revolution frequency, particle density ratio, particle size ratio, and drum length. After hyperparameter tuning by PSO on a validation set of 25 simulations, the SVR model was trained on 81 DEM simulations. Testing on another 25 simulations yielded excellent results with $R^2 = 0.95$ for mixing time and $R^2 = 0.90$ for mixing degree. These results indicate that the PSO-SVR model is suitable for rapid predictions of particle mixing in rotary drums and other particle processing equipment.

Keywords: Particle mixing, Rotary drum, DEM simulation, PSO-SVR,

*Corresponding author. Tel.: +49 391 67 52322.

Email address: wencong.wu@ovgu.de (W. Wu)

1. Introduction

Rotary drums are widely used for mixing granular materials in a variety of industries such as chemical, food, pharmaceutical, mineral, etc. Although the structure of a drum is quite simple, particle flow inside of the drum exhibits great complexity. According to the different operation conditions, particle motion in a rotary drum can evolve into six flow regimes: slipping, slumping, rolling, cascading, cataracting and centrifuging [1], where rolling and cascading regimes are commonly chosen for mixing operations since they provide relatively good particle movement [2–4]. Besides, when inventory particles have varying sizes or densities, segregation phenomena occur, smaller and denser particles generally tend to gather in the center of the particle bed [5]. In a over half filled drum at low revolution frequency, radial streaks may appear in a binary mixture system [5–7]. In a long rotary drum, a pattern of alternating bands can be observed in the axial direction because of size segregation [8–10]. Such complex particle dynamics have significant impacts on mixing time and mixing degree [11–13], which in turn affect the efficiency of a mixing process and the quality of final products. Therefore, prediction of particle mixing under different operation conditions is significant in industrial productions.

The discrete element method (DEM) developed by Cundall and Strack [14] is a powerful tool to investigate particle flow behavior [15–17], and has been widely used for particle mixing simulations [18–22]. Compared to the experiment, DEM simulation can provide detailed information for every par-

ticle in the whole system, including their instantaneous positions, instantaneous velocities and complete movement trajectories. These pieces of information are of great importance for further analysis of the particle flow, but are hard to obtain experimentally. However, large-scale industrial DEM modeling is still limited because of its extremely high computational cost [23–25], especially in automatic control applications where parameters are frequently changed.

In order to solve this problem, in recent years, some researchers turned their attention to artificial intelligence and tried to combine DEM simulations with data-driven models based on machine learning. For example, Kumar et al. [26] developed an artificial neural network model to predict the mass discharge rate from conical hoppers. Training data of this model were generated from a set of 67 DEM simulations with seven varied input parameters. Li et al. [27] used the support vector regression (SVR) method to predict two important properties of particle flow: angle of repose and collision energy inside of a drum. They selected a set of 84 DEM simulations based on Kennard-Stone algorithm to train their SVR model. Hu and Liu [28] proposed the Hermite interpolation-based Markov chains method to make predictions of the particle spatial distribution and mixing degree in a short rotary drum with periodic boundaries. Only six DEM simulations were used for training, but a simple monodisperse particle system with only one variable, the revolution frequency of the drum, was considered in this work.

Our previous work combined 2D DEM simulation results and a cross-correlation method to predict particle mixing in a 3D drum [29]. Although the cross-correlation can be developed based on a small amount of simulation

data and is able to make predictions beyond the parametric range, its application is limited by the number of changeable independent variables and the ability to only predict mixing time, not the mixing degree. In order to expand the predictive range to both, mixing time and mixing degree with more independent variables, in this work we develop a DEM data-driven machine learning model, which can be used for fast predictions of the particle mixing process of binary mixtures in rotary drums. DEM simulations will be performed by changing four different independent variables: revolution frequency of the drum, particle density ratio, particle size ratio and drum length, which means that complex segregation phenomena are also taken into account. The subdomain-based mixing index (SMI) is selected as the measure of mixedness. From the evolution of SMI, we can obtain two key characteristics of a mixing process: mixing time and mixing degree at steady mixing state. The next step is to import the data from DEM simulations into a machine learning model. Support vector regression (SVR) is combined with an algorithm of particle swarm optimization (PSO) for prediction. After the PSO-SVR model has been established, the mixing process within the whole parameter space can be predicted quickly, instead of performing additional time-consuming DEM simulations.

2. DEM simulation and data generation

2.1. Simulation conditions and parameters

The DEM simulations were performed for horizontal rotary drums with binary sphere mixtures. Two kinds of particles were generated in the upper half of the drum and then let to fall freely, finally forming a top-bottom

stratification, as shown in Fig. 1. White particles initially at the bottom (so-called benchmark particles) are always with constant size and density. The end walls can be considered to affect the size segregation pattern [10]. In order to investigate the influence of this segregation on mixing, we set the boundaries as fixed instead of periodic. The key parameters that maintain constant value in the DEM simulations are listed in Table 1. The current DEM simulation has been validated through experiments with a drum length of 150 mm as reported in our previous work [29].

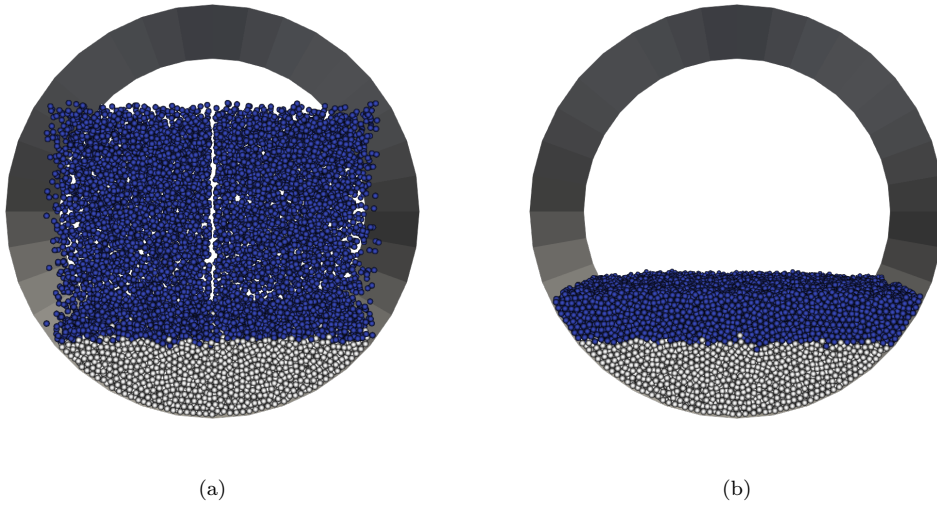


Figure 1: Particles in the 150 mm rotary drum with top-bottom stratification (18800 white particles + 18800 blue particles): (a) particle generation; (b) initial state before rotating.

In this work, we used the results from 131 DEM simulations (107 newly performed and 24 taken from our previous work) with four varied independent variables: revolution frequency of the drum, particle density ratio, particle size ratio and drum length. These 131 simulations are split into three sets: a training set (81), a validation set (25) and a test set (25). The training set is

Table 1: Key parameters in the DEM simulations with constant value.

Parameter	Value	Unit
Drum diameter	194	mm
Filling degree	0.2	-
Benchmark particle diameter	3	mm
Benchmark particle density	1000	kg/m ³
Young’s modulus	1×10^7	Pa
Poisson’s ratio	0.24	-
Coefficient of restitution	0.82	-
Friction coefficient	0.80	-
Rolling coefficient	0.01	-

used to train the model, the validation set is used for hyperparameter tuning, and the test set is used to evaluate the final performance of the chosen model. In the training set, every independent variable has three different levels, as shown in Table 2, resulting in 81 simulations. As for the validation set and the test set, the independent variables were chosen within the parametric range established by the training set, but with different values.

Table 2: Values of four independent variables in the training set.

Independent variable	Value		
Revolution frequency [rpm]	10	30	50
Particle density ratio [-]	1	1.5	2
Particle size ratio [-]	1	1.5	2
Drum length [mm]	75	150	300

We used the open-source software LIGGGHTS, which supports parallel computing, to conduct DEM simulations [30]. The particle number in the simulation is related to the drum length and the particle size ratio. Among the simulations conducted in this study, the maximum total particle number is reached in the 300 mm drum with a particle size ratio of one, amounting to 75200 particles (37600 + 37600). As the drum length decreases and the size ratio increases, the particle number diminishes proportionally. A typical simulation on a single core without parallel computing takes about 18.8 h for 60 s of mixing process time with 75200 particles; while using 16 cores for parallel computing, the time consumption is reduced to 5.2 h, i.e., by more than 3.6 times. All the simulations were performed on a commercial computer with AMD Ryzen Threadripper 2950X processor (16 cores/32 threads, 3.5 GHz) and 32 GB RAM.

2.2. DEM model

For an individual particle i with mass m_i in the DEM model, its translational and rotational motion at each time step can be described by Newton's second law as:

$$m_i \frac{d\mathbf{v}_i}{dt} = \sum_{j=1}^{N_c} (\mathbf{F}_{ij}^n + \mathbf{F}_{ij}^t) + m_i \mathbf{g}, \quad (1)$$

and

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_{j=1}^{N_c} (\mathbf{T}_{t,ij} + \mathbf{T}_{r,ij}), \quad (2)$$

where \mathbf{v} and $\boldsymbol{\omega}$ are the translational and rotational velocities of the particle, respectively. In addition to gravity, the contact force acting on particle i can be decomposed into normal and tangential directions. While interacting with

particle j , torque on particle i can also be divided into two parts: the tangential torque $\mathbf{T}_{t,ij}$, which makes the particle to rotate, and the rolling friction torque $\mathbf{T}_{r,ij}$, which impedes the relative rotation between two particles. Here, N_c represents the total number of particles contacting with particle i ; contact forces and torques from them are summarized. We chose the Hertz contact model with tangential loading history condition to describe particle interaction [31] and the constant directional torque model to calculate the torque contribution from the rolling friction [32].

2.3. Mixing quantification

A recently proposed mixing index, the subdomain-based mixing index (SMI) [33], was selected for the quantitative description of mixing in this study. Compared to other fraction-based indices, which have the distinct feature of calculating the number fraction of particles in selected regions, such as the classic Lacey index [34] and the widely used entropy of mixing [35], SMI is suitable for binary or multi-component mixtures mixed in any initial proportion. Moreover, the calculation of SMI includes the entire mixing system, avoiding the randomness caused by sampling.

The number of subdomains is kept the same as in our previous work (1250, corresponding to a grid of $5 \times 5 \times 50$). Specifically, in the axial direction, the rotary drum was first divided into 50 equivalent layers. Then a circumscribed square of the circular cross section of the drum was defined, and a 5×5 equidistant square grid was laid on this square. The detailed gridding manner is shown in Fig. 2.

The core idea of the SMI is obtaining local mixing information from all subdomains and then integrating them to get the overall index. For a mixture

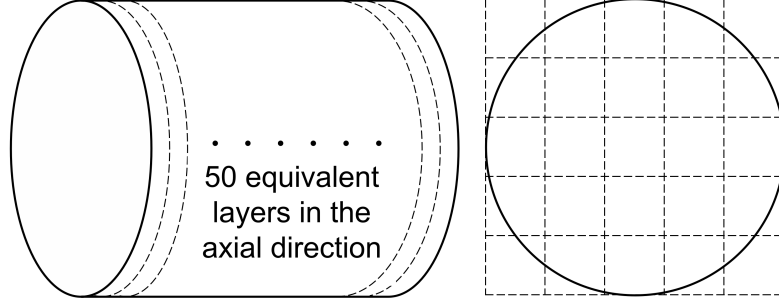


Figure 2: Gridding manner for SMI.

that consists of Q types of particles within n subdomains, the participation factor of particle type k can be calculated as:

$$f_k = \frac{\max(\sum_{i=1}^n N_{1i}, \sum_{i=1}^n N_{2i}, \dots, \sum_{i=1}^n N_{Qi})}{\sum_{i=1}^n N_{ki}}. \quad (3)$$

The modified fraction of k -type particles with respect to the maximum number of particles among the Q types in subdomain i is:

$$P_{ki} = \frac{N_{ki} f_k}{\max(N_{1i} f_1, N_{2i} f_2, \dots, N_{Qi} f_Q)}. \quad (4)$$

The local SMI for subdomain i is calculated as:

$$\text{SMI}(i) = \frac{1}{Q-1} \left(\sum_{k=1}^Q P_{ki} - 1 \right). \quad (5)$$

Finally, the total mixing index is obtained by averaging all the local SMIs as follows:

$$\text{SMI} = \frac{1}{N_{tot}} \sum_{i=1}^n \left[\text{SMI}(i) \sum_{k=1}^Q N_{ki} \right], \quad (6)$$

where N_{tot} is the total number of particles in the whole mixing system. The range of SMI is between zero and one: zero indicates a completely separated state, whereas one means that different types of particles are perfectly mixed.

After calculating the SMI values during a mixing process, the next step is to determine the mixing time and mixing degree at steady mixing state from the curve of SMI over time. In previous studies, researchers always applied different transformations of exponential functions or cubic polynomials to fit the index evolution [29, 36, 37]. However, this procedure is not suitable for mixing particles with different densities and diameters. Because when segregation phenomena occur, the shape of the mixing index evolution curve will dramatically change, especially for the size segregation, as shown in Fig. 3. The SMI curves all start from about 0.35 instead of zero and have big fluctuations during the first two seconds. At the initial state, the edges of the subdomains do not coincide with the dividing plane of the two colored layers; during the first turn, many particles change subdomain in a short time. When mixing particles with identical density and size, the SMI gradually increases and then enters a stable stage after a period of time, with finally quite well mixed particles, as can be seen in Fig. 3(a). If two kinds of particles have different density but same size, only density segregation occurs. The shape of the SMI curve here is similar to the first case, but it approaches the steady state much faster. As the mixing process proceeds, heavier blue particles tend to gather in the central axis of the bed and form a heavy core with lighter white particles surrounding, which leads to a low mixing degree, as illustrated in Fig. 3(b). Figure 3(c) shows the SMI evolution when there is only size segregation. After a short period of increasing, the mixing in the

rotary drum undergoes a transition from an unstable intermediate state to a stable segregation state, and the SMI drops sharply until it reaches a steady level, even smaller than the initial value, resulting in a prolonged mixing time and poor mixing quality. The bigger blue particles are collected at both ends of the drum, whereas smaller white particles are concentrated in the middle of the particle bed, together forming three distinct alternating bands. In Fig. 3(d) we can find that when density segregation and size segregation appear simultaneously, the SMI value decreases only slightly after having reached a peak. Compared with the previous two cases, here the mixing time becomes shorter and the steady SMI is also higher, which indicates that the two types of segregation oppose each other, facilitating mixing when simultaneously present. For a binary mixture, density segregation and size segregation tend to cancel each other when one type of the particles is larger than the other in both size and density, which has also been observed experimentally by Jain et al. [6].

In order to determine the mixing time and the mixing degree at steady state from different situations, we process the SMI evolution as follows:

- 1) Run each simulation for 60 s and check the difference in SMI between 50 s and 60 s. If the difference is less than 0.02, then average the SMI values over 50–60 s as the mixing degree at steady mixing state, otherwise continue the simulation for further 60 s and repeat the same procedure until steady state SMI has been found;
- 2) Calculate the moving average of the original SMI curve as the smoothed SMI curve by the smooth function in MATLAB with a span of the moving average of 49 considered data points to reduce noise or fluctuations;

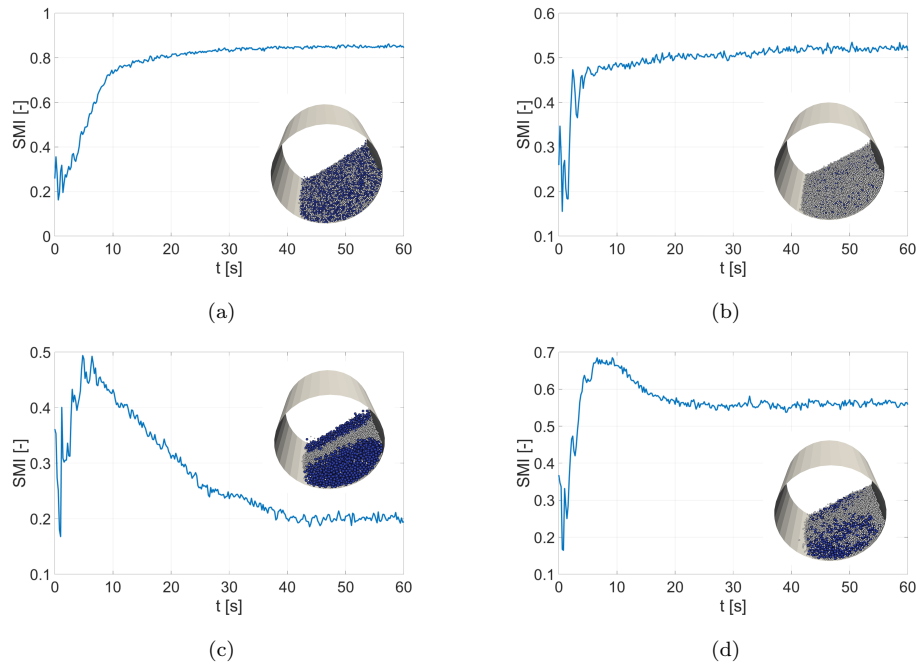


Figure 3: Different shapes of SMI evolution and corresponding mixing patterns at steady state (drum length = 150 mm, revolution frequency = 30 rpm), white particles are the benchmark particles with relatively low density and small size: (a) particle density ratio = 1, particle size ratio = 1, no segregation; (b) particle density ratio = 2, particle size ratio = 1, only density segregation; (c) particle density ratio = 1, particle size ratio = 2, only size segregation; (d) particle density ratio = 2, particle size ratio = 2, density and size segregation occur simultaneously.

- 3) Lay two horizontal straight lines from right to left on the image of the smoothed SMI curve, the values of which should correspond to 95% and 105% of the mixing degree at steady state respectively. Find the first intersection point between the curve and the lines, and take the corresponding time as the mixing time.

Figure 4 gives an example of the above mentioned determination process.

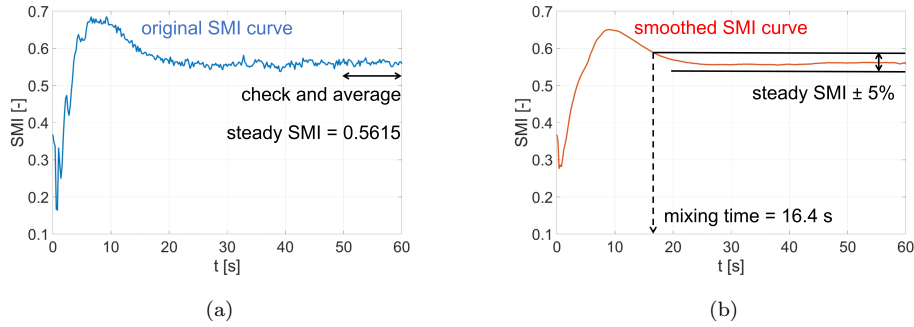


Figure 4: Example of determining (a) mixing degree at steady state; (b) mixing time from a SMI evolution.

2.4. Simulation results in the training set

The DEM simulation results for the training set are briefly analyzed in this section. Figure 5 plots the simplest cases, in which no segregation exists. In other words, the density ratio and the size ratio are both equal to one. The results about mixing time shown in Fig. 5(a) are generally in line with intuition: higher revolution frequencies lead to faster mixing, while longer rotary drums take longer to become well mixed. In the low revolution frequency region (10–30 rpm), increasing rotational speed has a significant effect on improving mixing efficiency. As for the mixing degree at steady mixing

state, Fig. 5(b) illustrates that the increase of revolution frequency has only little effect, but the mixing degree is relatively higher in longer drums. This is because for drums of different lengths the total number of subdomains in the SMI calculation is the same ($5 \times 5 \times 50 = 1250$). In shorter drums, non-uniformity may occur in the, absolutely seen, smaller subdomains, while the composition tends to become more uniform in the larger subdomains of the longer drum. This is actually a commonality of the fraction-based mixing indices. Both the authors of entropy of mixing [35] and SMI [33] have observed this phenomenon. When we choose a smaller subdomain, it means that we use a more rigorous perspective to examine the mixing state. In our previous study [29], subdomains of different sizes were also compared in more detail.

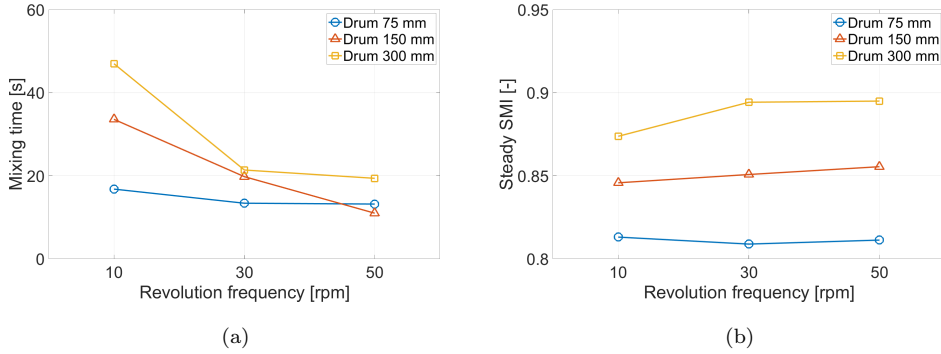


Figure 5: Relationship between mixedness and revolution frequency in drums of different lengths without segregation phenomenon: (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

Next cases where only density segregation occurs are considered. The variation of mixing time in Fig. 6(a) shows that, when there is a density segregation, increasing revolution frequency always leads to faster mixing. But in the high revolution frequency region (30–50 rpm), the mixing times

of particles with different density ratios are almost identical. Regarding the mixing degree at steady mixing state, unlike the case without segregation, increasing the revolution frequency significantly improves the mixing quality, as shown in Fig. 6(b). Additionally, the larger the density ratio, the poorer the mixing quality.

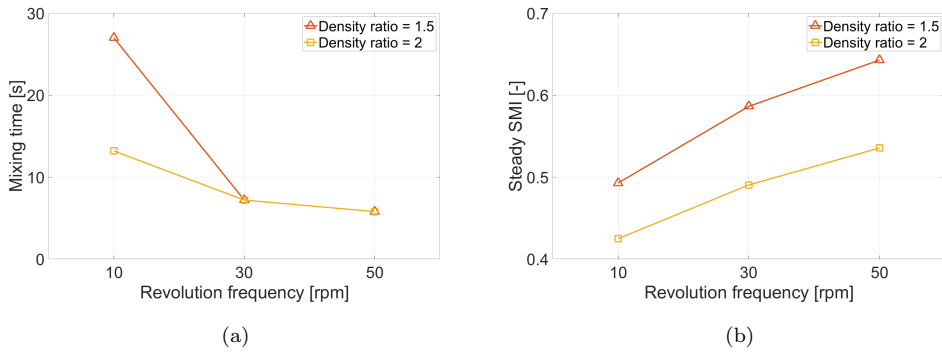


Figure 6: Relationship between mixedness and revolution frequency in the drum of 75 mm length with density ratio larger than one: (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

Finally, the cases where only size segregation appears are analyzed. Figure 7(a) shows that mixing time still follows the trend of faster mixing at higher revolution frequencies, but this trend is not that pronounced when mixing particles with high size ratio (2) in the high revolution frequency region (30–50 rpm). It also can be seen that particles with larger size ratio are mixed (or rather segregated) much faster than those with smaller size ratio. The behavior of mixing degree at steady mixing state can be seen in Fig. 7(b). Similar to the case of density segregation, when size segregation occurs, increasing the revolution frequency promotes mixing. However, the difference in steady mixing degree between a particle size ratio of 1.5 and

2 is negligible. This maybe because a particle size ratio of 1.5 is enough to form clear and stable segregation bands in the axial direction, so that further increasing the particle size ratio does not significantly deteriorate the mixing degree.

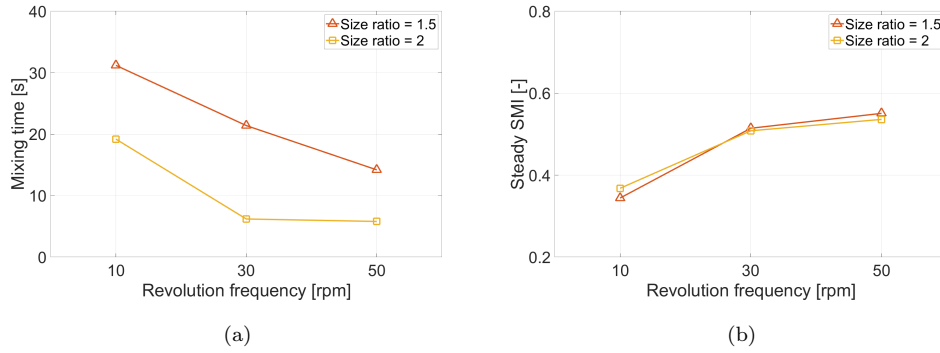


Figure 7: Relationship between mixedness and revolution frequency in the drum of 75 mm length with size ratio larger than one: (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

From the above analysis, it can be observed that each independent variable (revolution frequency, particle density ratio, particle size ratio, drum length) has its own distinct effects on mixing time and mixing degree. When they interact together, the situation becomes much more complex, making it extremely challenging to capture the intricate relationships and nonlinearities between the parameters and the mixing outcomes through conventional empirical methods. In this kind of situation, the use of machine learning methods may be highly rewarding.

3. Data-driven model for prediction

3.1. Support vector regression model

Support vector regression (SVR) is a variation that extends the use of support vector machines from solving classification problems to solving regression problems [38, 39]. Compared to other machine learning models, SVR has outstanding advantages in handling non-linear and high-dimensional data, especially with small sample size. Therefore, we chose SVR as the DEM data-driven model for mixing prediction in this work.

The basic idea behind SVR while dealing with non-linear regression is: first, transforming the non-linear problem into linear by mapping samples from the input space to a high-dimensional feature space; then constructing an optimal hyperplane in the feature space, which minimizes the distance between partial training samples (so-called support vectors) and the hyperplane, while still allowing a certain degree of error on both sides of the hyperplane. Those two processes are illustrated in Fig. 8, with mathematical symbols that will be explained in the following.

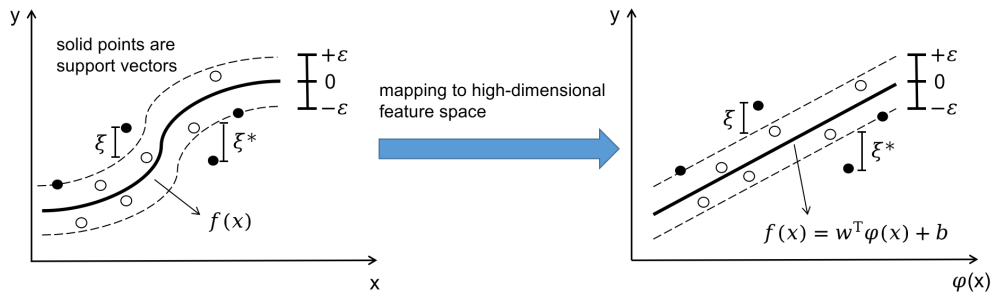


Figure 8: Scheme of SVR for non-linear regression in one dimension.

Specifically, assume training set data to be $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\}$, where l

is the number of samples in the training set, which is equal to 81 in the present work. Our input data vector \mathbf{x}_i includes four elements: revolution frequency, particle density ratio, particle size ratio, and drum length, whereas the target value y represents either mixing time or mixing degree at steady mixing state, calculated based on the DEM simulation results. The hyperplane in the high-dimensional feature space can be represented as:

$$f(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + b, \quad (7)$$

where \mathbf{w} is the vector normal to the hyperplane, b denotes the offset, and $\varphi(\mathbf{x})$ represents the non-linear mapping function which maps the original input data to the high-dimensional feature space.

The goal of SVR is not only to minimize the error between predicted and actual values but also to ensure that the function is as simple as possible. Therefore, the objective function of SVR can be expressed as [40]:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^l \ell_{\epsilon}(f(\mathbf{x}_i) - y_i), \quad (8)$$

where the first term is the regularization term of the model, which controls the complexity of the model; the second term represents the loss function of the training error; and the penalty coefficient C is one of the hyperparameters used to balance the fitting ability and generalization ability of the model. A smaller C allows the model to be more tolerant upon training errors. On the other hand, a larger C imposes a higher penalty for errors, leading to a stricter fit to the training data, which may result in overfitting. It should be noted that this kind of hyperparameter needs to be set in advance before training.

Unlike the commonly used least square method, which calculates the prediction error for every data point, SVR adopts an ϵ -insensitive loss function, only penalizing predictions that are farther than ϵ from the desired output:

$$\ell_{\epsilon}(a) = \begin{cases} 0, & \text{if } |a| \leq \epsilon \\ |a| - \epsilon, & \text{otherwise.} \end{cases} \quad (9)$$

In Fig. 8, this ϵ -insensitive loss function can be visualized as a tube. Data points that fall within this tube have a loss of zero, since their predictions are within an acceptable range. Data points which lie on the edge of the tube or outside the tube are called support vectors. A larger ϵ means a wider tube as well as less support vectors, which may lead to more training error but can result in a model that is insensitive to noise. A small ϵ will lead a narrower tube with more support vectors outside, which makes the model more accurate on the training data but also increases the risk of overfitting. Similar to the penalty coefficient C , the margin of tolerance ϵ is another hyperparameter that needs to be set before training the SVR model.

By introducing the slack variables ξ_i and ξ_i^* , which represent the distance between support vectors and the upper and lower margin of the ϵ -insensitive tube, Eq. 8 can be reformulated as:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*), \quad (10)$$

subject to:

$$\begin{aligned} y_i - f(\mathbf{x}_i) &\leq \epsilon + \xi_i, \\ f(\mathbf{x}_i) - y_i &\leq \epsilon + \xi_i^*, \\ \xi_i, \xi_i^* &\geq 0, i = 1, 2, \dots, l. \end{aligned}$$

By constructing a Lagrange function, the relationship between normal vector of the hyperplane \mathbf{w} and non-linear mapping function $\varphi(\mathbf{x})$ can be derived as:

$$\mathbf{w} = \sum_{i=1}^l (\alpha_i - \alpha_i^*) \varphi(\mathbf{x}_i), \quad (11)$$

where α_i and α_i^* are Lagrange multipliers. Substituting Eq. 11 into Eq. 7, we get:

$$f(\mathbf{x}) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}) + b = \sum_{i=1}^l (\alpha_i - \alpha_i^*) k(\mathbf{x}_i, \mathbf{x}) + b, \quad (12)$$

where $k(\mathbf{x}_i, \mathbf{x})$ is the kernel function, the result of which, calculated in the original input space, is equal to the inner product of the transformed feature vectors $\varphi(\mathbf{x}_i)^T \varphi(\mathbf{x})$. The introduction of the kernel function is motivated by the inherent difficulty in directly computing the inner product in high-dimensional feature spaces. By applying the kernel function, we no longer need to know the specific form of the mapping function $\varphi(\mathbf{x})$, because the high-dimensional feature space can be implicitly defined by specifying the kernel function.

For the non-linear SVR, there are many different kernel functions to choose from, such as polynomial kernel, radial basis function (RBF) kernel and sigmoid kernel. Among them, the RBF kernel function was selected in this work because it offers a good balance between flexibility, computational efficiency, and ease of use. Its expression is:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \quad (13)$$

where γ is the kernel function coefficient, which needs to be greater than zero. And, this is the last hyperparameter in our SVR model.

Additionally, since our four input variables (revolution frequency of the drum, particle density ratio, particle size ratio, drum length) vary in different ranges, preprocessing is necessary. We standardize each value of every input variables by subtracting the respective mean and scaling to unit variance, which can be expressed as:

$$\mathbf{x}_{i,st} = \frac{\mathbf{x}_i - \mathbf{u}}{\mathbf{s}}, \quad (14)$$

where $\mathbf{x}_{i,st}$ is the standard score of the input vector \mathbf{x}_i , \mathbf{u} is the mean of this vector in the training sample, and \mathbf{s} is its standard deviation.

3.2. Particle swarm optimization

As mentioned in the last section, there are three hyperparameters: penalty coefficient C , margin of tolerance ϵ and kernel function coefficient γ in the SVR model, which are not learned from the training data but need to be set before training in order to control the behavior of the learning algorithm. Setting hyperparameters correctly is crucial for achieving good performance with the predictive model.

In Section 2.1, we explained that all of the 131 simulation data were divided into three sets, where the validation set which consists of 25 simulations was prepared for hyperparameter tuning. Hyperparameter tuning for SVR is essentially an optimization problem with the objective of finding the best set of hyperparameters that gives the highest predictive accuracy on the validation set. Particle swarm optimization (PSO) is a metaheuristic optimization algorithm inspired by the social behavior of bird flocking or fish schooling [41]. This optimization method is able to scan the entire search

space and give the global optimal solution efficiently, which makes it really suitable for hyperparameter tuning in SVR.

The first thing to note is that the so-called particles in PSO are different from the particles mixing in the drum. In PSO, a group of particles, also called the swarm, each representing a potential solution, move through the search space towards the optimum solution by updating their positions and velocities based on their own experiences and the experiences of their neighboring particles. The algorithm begins with a random initialization of particle positions within the search space. Each particle evaluates its position using an objective function and keeps track of its personal best solution. The particles then update their velocities by considering their personal best positions and the global best position found by any particle in the swarm. The particles' positions are also updated based on their velocity. Through this iterative process, PSO explores the search space and converges towards the global optimum efficiently.

The objective function here evaluates the performance of each hyperparameter setting on the validation set, which can be characterized by the coefficient of determination:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^l (y_i - \hat{y}_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2}, \quad (15)$$

where \hat{y}_i is the predicted value of mixing time or mixing degree at steady mixing state from SVR model of the i -th sample, y_i corresponds to the true value obtained from DEM simulation results and \bar{y} is the mean of true values across all l samples. The R^2 value indicates the proportion of the variance in the true values that is explained by the predicted values, with higher values indicating better predictive performance.

In PSO, for particle i at iteration r , the velocity and position updates are given by [42]:

$$\mathbf{v}_{i,r+1} = w\mathbf{v}_{i,r} + c_1r_1(\mathbf{p}_{i,r}^{best} - \mathbf{p}_{i,r}) + c_2r_2(\mathbf{g}_r^{best} - \mathbf{p}_{i,r}), \quad (16)$$

$$\mathbf{p}_{i,r+1} = \mathbf{p}_{i,r} + \mathbf{v}_{i,r+1}, \quad (17)$$

where $\mathbf{p}_{i,r} = (C, \epsilon, \gamma)$ is the position vector of particle i at iteration r , $\mathbf{v}_{i,t}$ is the corresponding velocity vector, $\mathbf{p}_{i,r}^{best}$ is the personal best position found by particle i up to iteration r , \mathbf{g}_r^{best} is the global best position found by the swarm up to iteration r , w is the inertia weight that controls the impact of the previous velocity on the current velocity, c_1 and c_2 are the acceleration coefficients that control the impact of the personal best and global best positions on the particle's velocity, r_1 and r_2 are random numbers sampled from a uniform distribution between zero and one. In order to balance exploration and exploitation of the search space, the inertia weight w is set as 0.4, and the setting of acceleration coefficients is $c_1 = c_2 = 2$.

3.3. PSO-SVR model implementation

Our PSO-SVR model is written in Python. The SVR part has been implemented through an open source machine learning library, Scikit-learn [43], and then connected to the self-programmed PSO model. The flowchart of the whole model is shown on Fig. 9.

As mentioned in Section 2.1, the DEM simulation data were first grouped into three sets, where the validation set was imported into the PSO for hyperparameter tuning. The PSO model starts with initialization of population parameters, which include the population size and the number of generations.

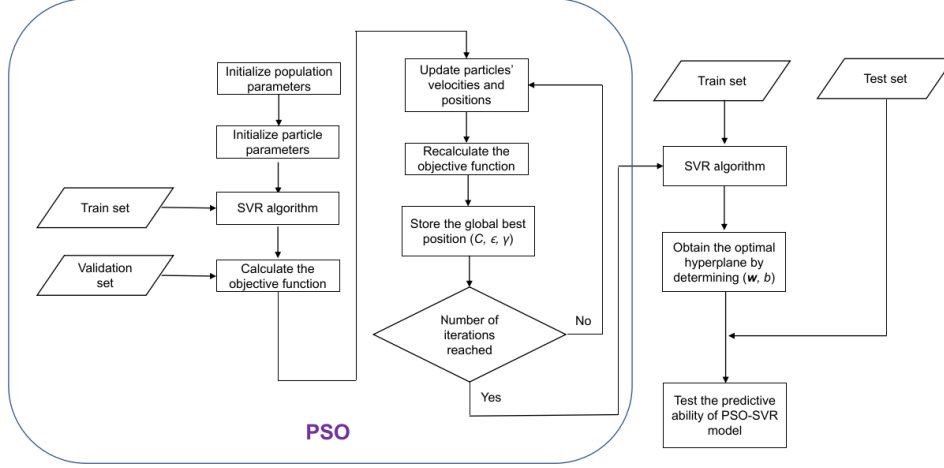


Figure 9: Flowchart of PSO-SVR model.

The population size is the number of particles (solutions) in PSO, and the number of generations is the maximum number of iterations. We observed that varying those two population parameters from 20 to 40 gives essentially the same results, so both of them are set as 20. Then, after initializing the particles' velocities and positions, the objective function (Eq. 15) can be calculated through the SVR algorithm. For a given combination of hyperparameters (C, ϵ, γ) , the SVR algorithm will compute a present optimal hyperplane with fixed (\mathbf{w}, b) (Eq. 7). As the cycle proceeds, the values of (C, ϵ, γ) and (\mathbf{w}, b) will be updated continuously, until the maximum number of iterations has been reached. Eventually, the global best position (C, ϵ, γ) of PSO in the last iteration will be stored and imported into the SVR algorithm along with the training set for PSO-SVR model training, and the final optimal hyperplane will be obtained. This marks the fact that the PSO-SVR

model has been established and can be used to predict new data (test set).

It is worth noting that we developed two PSO-SVR models for predicting mixing time and mixing degree at steady mixing state separately, to ensure their predictive accuracy. In other words, the combinations of hyperparameters in these two predictive models are different, which reflects the specific requirements of each prediction task, resulting in models that are tailored to their respective objectives. Through PSO, the optimal hyperparameters for mixing time prediction are determined as: $(C, \epsilon, \gamma) = (67.8, 0.1, 1.71)$, and for mixing degree prediction as: $(C, \epsilon, \gamma) = (16.1, 0.001, 0.151)$. It is interesting to see that the margin of tolerance (ϵ) of SVR model for mixing time is much larger than that for mixing degree, indicating that the model for mixing time is more tolerant to variations and uncertainties. This may be because the ranges of those output variables are different. In the validation set, the maximum and minimum values of the mixing degree only differ by three times, but that of mixing time can differ by more than six times.

4. Predictive results from PSO-SVR model

After hyperparameter tuning, the best set of hyperparameters was used to train the predictive model using the training set. The final performance of the model was then evaluated on the test set. It is noteworthy that the PSO-SVR model exhibits a remarkable improvement in time efficiency, as it only takes tens of seconds from hyperparameter tuning to prediction, compared to spending several hours to run a new DEM simulation. The predictive results of mixing time and mixing degree at steady mixing state in the training and test set are summarized by means of parity plots in Fig. 10.

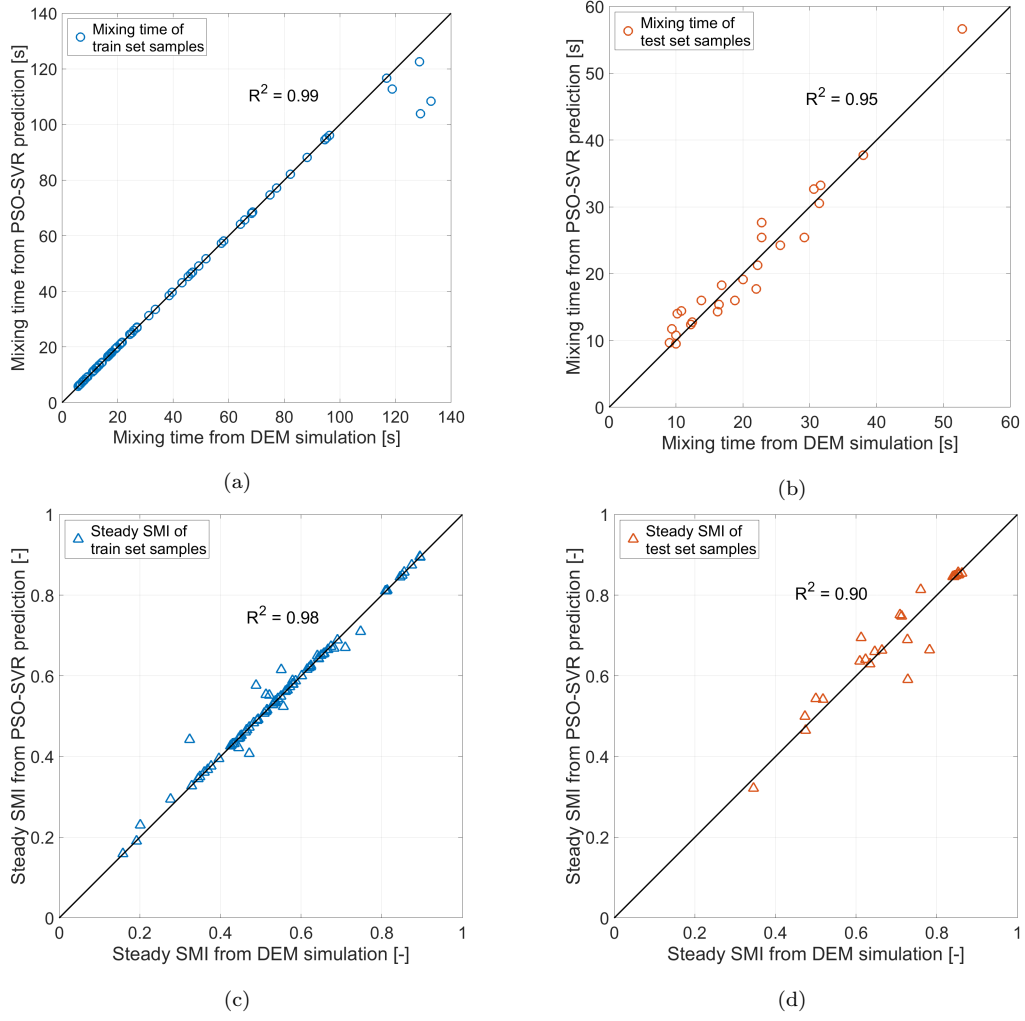


Figure 10: Comparison between DEM simulation and PSO-SVR prediction: (a) mixing time of the train set; (b) mixing time of the test set; (c) mixing degree (SMI) at steady mixing state of the train set; (d) mixing degree (SMI) at steady mixing state of the test set.

On the whole, the prediction of mixing time in the training set has an extremely high accuracy of $R^2 = 0.99$, as can be seen in Fig. 10(a). Among a total of 81 training data points, only four points deviate from the reference line due to under-forecasting. These four mixing processes all appear under the lowest revolution frequency (10 rpm) with relatively large particle size ratio (1.5 or 2) in relatively long drum (150 or 300 mm). The mixing curves under these conditions are distinct from a lot of other cases, similar to the curve shown in Fig. 3(c). The drum in Fig. 3(c) is, though, 150 mm long, and in the still longer drum, the alternating bands in the axial direction will be more numerous and unstable, as shown in Fig. 11, further increasing the difficulty in predicting the mixing time. However, such error can be reduced by adding more similarly conditioned data points to the training set.

As for the mixing time in the test set (Fig. 10(b)), the PSO-SVR model provides very satisfactory result with $R^2 = 0.95$, which is only slightly lower than its predictive accuracy on the training set, shows the strong generalization ability of this model. And, the data points are evenly distributed on both sides of the reference line, which indicates that the model does not exhibit any particular overestimation or underestimation tendencies.

Figures 10(c) and (d) show the prediction of mixing degree (SMI) at steady mixing state in the training set and the test set, respectively. The predictive accuracy of the PSO-SVR model for the training set is quite good with $R^2 = 0.98$. The predictive accuracy for the test set remains acceptable ($R^2 = 0.90$), though visibly smaller. The reason can be found in Table 3, which gives the detailed predicted results of the test set.

To present the results more intuitively, different measures were adopted

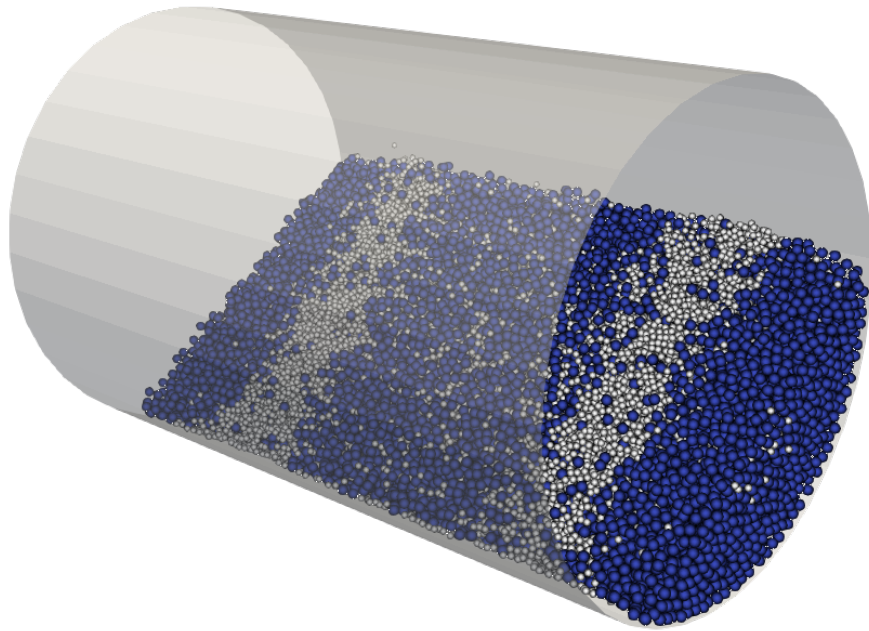


Figure 11: Alternating bands in the 300 mm drum, blue particles have the bigger size (revolution frequency = 10 rpm, particle density ratio = 1, particle size ratio = 1.5).

Table 3: Prediction results of test set.

Independent variable				Mixing time			Mixing degree at steady mixing state		
n	r_{dens}	r_{size}	L	t_{DEM}	$t_{\text{PSO-SVR}}$	t_{AE}	SMI_{DEM}	$\text{SMI}_{\text{PSO-SVR}}$	SMI_{APE}
[rpm]	[-]	[-]	[mm]	[s]	[s]	[s]	[-]	[-]	[%]
40	2	1.5	150	16.4	15.4	1.0	0.664	0.664	0.0%
30	1.25	1.75	150	38.0	37.7	0.3	0.346	0.321	7.0%
11	1	1	150	31.6	33.2	1.6	0.843	0.846	0.3%
12.5	1	1	150	30.6	32.7	2.1	0.839	0.846	0.8%
22.5	1	1	150	22.8	25.4	2.6	0.844	0.848	0.5%
27.5	1	1	150	22.2	21.3	0.9	0.848	0.849	0.1%
35	1	1	150	16.8	18.3	1.5	0.856	0.852	0.5%
40	1	1	150	13.8	16.0	2.2	0.862	0.854	0.9%
45	1	1	150	12.2	12.4	0.2	0.853	0.856	0.4%
10	1.75	1	150	12.4	12.7	0.3	0.474	0.465	2.0%
20	1.75	1	150	18.8	16.0	2.8	0.518	0.541	4.5%
40	1.75	1	150	16.2	14.3	1.9	0.609	0.637	4.5%
50	1.75	1	150	10.0	9.5	0.5	0.634	0.630	0.7%
32	1	1	150	20.0	19.2	0.8	0.854	0.850	0.4%
16	1.8	1	150	10.2	14.0	3.8	0.473	0.500	5.6%
48	1.4	1	150	9.0	9.7	0.7	0.713	0.748	4.8%
20	1.25	1.2	100	29.2	25.4	3.8	0.728	0.591	18.8%
20	1.25	1	100	22.0	17.7	4.3	0.612	0.694	13.5%
20	1.75	1.6	100	22.8	27.6	4.8	0.624	0.641	2.8%
40	1.25	1.2	100	10.8	14.4	3.6	0.727	0.689	5.2%
40	1.25	1	100	10.0	10.8	0.8	0.709	0.752	6.0%
40	1.75	1.6	100	9.4	11.7	2.3	0.646	0.660	2.2%
20	1.75	1.6	200	52.8	56.6	3.8	0.500	0.543	8.7%
40	1.25	1.2	200	31.4	30.5	0.9	0.782	0.664	15.0%
40	1.25	1	200	25.6	24.2	1.4	0.760	0.814	7.1%

* n is the revolution frequency of the drum, r_{dens} is the particle density ratio, r_{size} is the particle size ratio, L is the drum length, AE means absolute error and APE represents absolute percentage error, both as absolute (always positive) values.

to depict the errors in the two output variables. Specifically, the absolute error was used to characterize the deviation of mixing time, while the percentage error was used to quantify the deviation of mixing degree at steady mixing state; both as absolute values, i.e. always positive defined. It can be found that, in terms of mixing degree, the three sets of data with the largest percentage errors exceeding 10% were all generated at the density ratio of 1.25. Notably, when the density ratio equals to 1, there is no density segregation, whereas a density ratio of 1.5 leads to significant segregation. The lack of data describing intermediate transition states in the training set results in less accurate predictions of the mixing degree. Choosing smaller intervals of density ratio variation in the train set could potentially enhance the predictive accuracy in this regard. Nevertheless, in general, the mean percentage error of mixing degree prediction in the test set is only 4.5%, which proves the effectiveness of the PSO-SVR model. Moreover, as for the mixing time in the test set, the mean absolute error is 2.0 s and the maximum absolute error is 4.8 s. Such small errors indicate that the PSO-SVR model is also highly suitable for predicting mixing times.

5. Conclusion

We developed a data-driven PSO-SVR model based on the results of DEM simulations to rapidly predict the particle mixing process, including mixing time and mixing degree at the steady mixing state, for binary mixtures in rotary drums. Four independent variables: revolution frequency of the drum, particle density ratio, particle size ratio and drum length were taken into account.

The prediction results of mixing time and mixing degree at steady mixing state in the training set are excellent with $R^2 = 0.99$ and $R^2 = 0.98$, respectively. Very satisfactory results ($R^2 = 0.95$) are obtained for the mixing time prediction in the test set, with a mean absolute difference to the real value of only 2.0 s. Although the predictive accuracy of mixing degree at steady mixing state in the test set is lower with $R^2 = 0.90$, this value and the small mean percentage error (4.5%) still make the result acceptable.

Overall, the present work demonstrates the effectiveness and efficiency of using the PSO-SVR model for predicting particle mixing in a rotary drum. This lays a foundation for extending the model to larger and more complex industrial mixers. In future work, we aim to consider additional independent variables, such as different filling degrees and particle shapes, and compare the PSO-SVR model with other data-driven machine learning models to further improve prediction accuracy.

Acknowledgments

The authors would like to gratefully acknowledge the funding of this work by the German Federal Ministry of Science and Education (BMBF) within the project ADMIX (03INT609BA).

Nomenclature

Symbols

a	loss of prediction	-
b	offset	-
c_1 and c_2	acceleration coefficients	-

C	penalty coefficient	-
f_k	participation factor of particle type k	-
\mathbf{F}	force	N
\mathbf{g}	acceleration of gravity	m/s ²
\mathbf{g}^{best}	global best position	-
I	moment of inertia	kg/m ²
k	kernel function	-
l	number of samples	-
ℓ_ϵ	ϵ loss function	-
m	mass	kg
n	number of subdomains	-
N	number of particles	-
\mathbf{p}	position vector	-
\mathbf{p}^{best}	personal best position	-
P	intermediate fraction in SMI	-
Q	types of particles	-
r_1 and r_2	random numbers between zero and one	-
R^2	coefficient of determination	-
\mathbf{s}	standard deviation of the input vector	-
t	time	s
\mathbf{T}_t	tangential torque	N · s
\mathbf{T}_r	rolling friction torque	N · s
\mathbf{u}	mean of the input vector	-
\mathbf{v}	velocity vector in PSO	-
w	inertia weight	-

\mathbf{w}	normal vector of the hyperplane	-
\mathbf{x}	input vector	-
\mathbf{x}_{st}	standard score of the input vector	-
y	target value	-
\hat{y}	predicted value	-
\bar{y}	mean of target values	-
Greek symbols		
α and α^*	Lagrange multipliers	-
γ	kernel function coefficient	-
ϵ	margin of tolerance	-
ξ and ξ^*	slack variables	-
ϕ	non-linear mapping function	-
Superscripts and subscripts		
c	contacting	
i, j, k	indices	
n	normal direction	
r	iteration	
t	tangential direction	
T	transpose	
tot	total	
Abbreviations		
DEM	discrete element method	
PSO	particle swarm optimization	
SMI	subdomain-based mixing index	
SVR	support vector regression	

References

- [1] H. Henein, J. Brimacombe, A. Watkinson, Experimental study of transverse bed motion in rotary kilns, *Metallurgical Transactions B* 14 (2) (1983) 191–205.
- [2] J. Mellmann, The transverse motion of solids in rotating cylinders: Forms of motion and transition behavior, *Powder Technology* 118 (3) (2001) 251–270.
- [3] A. A. Aissa, C. Duchesne, D. Rodrigue, Transverse mixing of polymer powders in a rotary cylinder, Part I: Active layer characterization, *Powder Technology* 219 (2012) 193–201.
- [4] L. F. Orozco, J.-Y. Delenne, P. Sornay, F. Radjai, Rheology and scaling behavior of cascading granular flows in rotating drums, *Journal of Rheology* 64 (4) (2020) 915–931.
- [5] D. Khakhar, A. V. Orpe, S. Hajra, Segregation of granular materials in rotating cylinders, *Physica A: Statistical Mechanics and its Applications* 318 (1-2) (2003) 129–136.
- [6] N. Jain, J. M. Ottino, R. M. Lueptow, Regimes of segregation and mixing in combined size and density granular systems: An experimental study, *Granular Matter* 7 (2) (2005) 69–81.
- [7] C. Liao, S. Hsiau, H. Nien, Density-driven spontaneous streak segregation patterns in a thin rotating drum, *Physical Review E* 89 (6) (2014) 062204.

- [8] K. Hill, A. Caprihan, J. Kakalios, Axial segregation of granular media rotated in a drum mixer: Pattern evolution, *Physical Review E* 56 (4) (1997) 4386.
- [9] H. Kuo, R. Hsu, Y. Hsiao, Investigation of axial segregation in a rotating drum, *Powder Technology* 153 (3) (2005) 196–203.
- [10] Z. Cui, Y. Zhao, Y. Chen, X. Liu, Z. Hua, C. Zhou, J. Zheng, Transition of axial segregation patterns in a long rotating drum, *Particuology* 13 (2014) 128–133.
- [11] M. Arntz, W. K. den Otter, W. J. Briels, P. Bussmann, H. Beftink, R. Boom, Granular mixing and segregation in a horizontal rotating drum: A simulation study on the impact of rotational speed and fill level, *AIChE Journal* 54 (12) (2008) 3133–3146.
- [12] H. Chen, X. Zhao, Y. Xiao, Y. Liu, Y. Liu, Radial mixing and segregation of granular bed bi-dispersed both in particle size and density within horizontal rotating drum, *Transactions of Nonferrous Metals Society of China* 26 (2) (2016) 527–535.
- [13] R. J. Brandao, R. M. Lima, R. L. Santos, C. R. Duarte, M. A. Barrozo, Experimental study and DEM analysis of granular segregation in a rotating drum, *Powder Technology* 364 (2020) 1–12.
- [14] P. A. Cundall, O. D. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1) (1979) 47–65.
- [15] Z. Jiang, C. Rieck, A. Bück, E. Tsotsas, Modeling of inter- and intra-particle coating uniformity in a Wurster fluidized bed by a coupled CFD-

- DEM-Monte Carlo approach, *Chemical Engineering Science* 211 (2020) 115289.
- [16] L. Zhang, Z. Jiang, F. Weigler, F. Herz, J. Mellmann, E. Tsotsas, PTV measurement and DEM simulation of the particle motion in a flighted rotating drum, *Powder Technology* 363 (2020) 23–37.
 - [17] L. Zhang, Z. Jiang, J. Mellmann, F. Weigler, F. Herz, A. Bück, E. Tsotsas, Influence of the number of flights on the dilute phase ratio in flighted rotating drums by PTV measurements and DEM simulations, *Particuology* 56 (2021) 171–182.
 - [18] M. Kwapinska, G. Saage, E. Tsotsas, Mixing of particles in rotary drums: A comparison of discrete element simulations with experimental results and penetration models for thermal processes, *Powder Technology* 161 (1) (2006) 69–78.
 - [19] P. Liu, R. Yang, A. Yu, DEM study of the transverse mixing of wet particles in rotating drums, *Chemical Engineering Science* 86 (2013) 99–107.
 - [20] Y. Wen, M. Liu, B. Liu, Y. Shao, Comparative study on the characterization method of particle mixing index using DEM method, *Procedia Engineering* 102 (2015) 1630–1642.
 - [21] X. Liu, Z. Hu, W. Wu, J. Zhan, F. Herz, E. Specht, DEM study on the surface mixing and whole mixing of granular materials in rotary drums, *Powder Technology* 315 (2017) 438–444.

- [22] M. K. Saeed, M. S. Siraj, Mixing study of non-spherical particles using DEM, *Powder Technology* 344 (2019) 617–627.
- [23] N. Govender, R. Rajamani, D. N. Wilke, C.-Y. Wu, J. Khinast, B. J. Glasser, Effect of particle shape in grinding mills using a GPU based DEM code, *Minerals Engineering* 129 (2018) 71–84.
- [24] B. Yan, R. Regueiro, Large-scale dynamic and static simulations of complex-shaped granular materials using parallel three-dimensional discrete element method (DEM) on DoD supercomputers, *Engineering Computations* 35 (2) (2018) 1049–1084.
- [25] C. Coetzee, Particle upscaling: Calibration and validation of the discrete element method, *Powder Technology* 344 (2019) 487–503.
- [26] R. Kumar, C. M. Patel, A. K. Jana, S. R. Gopireddy, Prediction of hopper discharge rate using combined discrete element method and artificial neural network, *Advanced Powder Technology* 29 (11) (2018) 2822–2834.
- [27] Y. Li, J. Bao, A. Yu, R. Yang, A combined data-driven and discrete modelling approach to predict particle flow in rotating drums, *Chemical Engineering Science* 231 (2021) 116251.
- [28] Z. Hu, X. Liu, A novel Markov chain method for predicting granular mixing process in rotary drums under different rotation speeds, *Powder Technology* 386 (2021) 40–50.
- [29] W. Wu, K. Chen, E. Tsotsas, Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation, *Advanced Powder Technology* 33 (4) (2022) 103512.

- [30] C. Kloss, C. Goniva, A. Hager, S. Amberger, S. Pirker, Models, algorithms and validation for opensource DEM and CFD-DEM, *Progress in Computational Fluid Dynamics, An International Journal* 12 (2-3) (2012) 140–152.
- [31] A. Di Renzo, F. P. Di Maio, Comparison of contact-force models for the simulation of collisions in DEM-based granular flow codes, *Chemical Engineering Science* 59 (3) (2004) 525–541.
- [32] J. Ai, J.-F. Chen, J. M. Rotter, J. Y. Ooi, Assessment of rolling resistance models in discrete element simulations, *Powder Technology* 206 (3) (2011) 269–282.
- [33] M. Cho, P. Dutta, J. Shim, A non-sampling mixing index for multicomponent mixtures, *Powder Technology* 319 (2017) 434–444.
- [34] P. M. C. Lacey, Developments in the theory of particle mixing, *Journal of Applied Chemistry* 4 (5) (1954) 257–268.
- [35] M. Schutyser, J. Padding, F. Weber, W. Briels, A. Rinzema, R. Boom, Discrete particle simulations predicting mixing behavior of solid substrate particles in a rotating drum fermenter, *Biotechnology and Bioengineering* 75 (6) (2001) 666–675.
- [36] G. Finnie, N. P. Kruyt, M. Ye, C. Zeilstra, J. Kuipers, Longitudinal and transverse mixing in rotary kilns: A discrete element method approach, *Chemical Engineering Science* 60 (15) (2005) 4083–4091.

- [37] Z. Zuo, S. Gong, G. Xie, J. Zhang, Sensitivity analysis of process parameters for granular mixing in an intensive mixer using response surface methodology, *Powder Technology* 384 (2021) 51–61.
- [38] C. Cortes, V. Vapnik, Support-vector networks, *Machine Learning* 20 (1995) 273–297.
- [39] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, V. Vapnik, Support vector regression machines, *Advances in Neural Information Processing Systems* 9 (1996).
- [40] C.-C. Chang, C.-J. Lin, Libsvm: A library for Support Vector Machines, *ACM Transactions on Intelligent Systems and Technology (TIST)* 2 (3) (2011) 1–27.
- [41] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *Proceedings of ICNN'95, International Conference on Neural Networks*, Vol. 4, IEEE, 1995, pp. 1942–1948.
- [42] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, IEEE, 1998, pp. 69–73.
- [43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.

Chapter 4

Prediction of rod-like particle mixing in rotary drums by three machine learning methods based on DEM simulation data

A machine learning model has been successfully applied in Chapter 3 to predict the mixing of spherical particles in rotary drums, yielding highly satisfactory results. Building on this success, the approach was extended to predict the mixing of non-spherical particles. A typical non-spherical particle type—rod-like particles—was selected and represented using the multi-sphere method. Additionally, three different machine learning models were developed and comprehensively compared.

A paper with the same title as this chapter has already been published:

Wencong Wu, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of rod-like particle mixing in rotary drums by three machine learning methods based on DEM simulation data.” *Powder Technology* 448 (2024): 120307.

This article is open access. Therefore, in accordance with the CC BY-NC 4.0 license (<https://creativecommons.org/licenses/by-nc/4.0/>), the published electronic version is provided, corresponding to pages 133–146 of this dissertation. It can also be accessed via the following link:

<https://doi.org/10.1016/j.powtec.2024.120307>.

As the first author, my **individual contributions** include: designing and performing all DEM simulations, analyzing the simulation data, developing the three machine learning models—ANN, ERT and PSO-SVR for mixing prediction, and writing the original draft with all figures and tables.



Prediction of rod-like particle mixing in rotary drums by three machine learning methods based on DEM simulation data

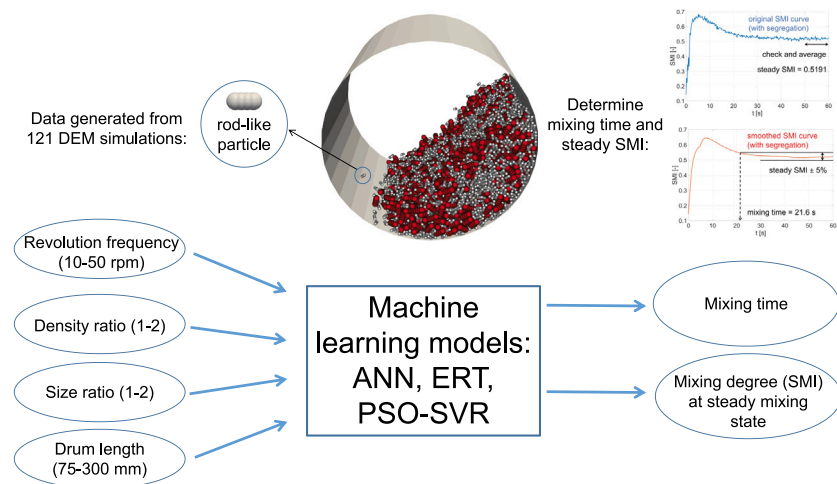
Wencong Wu^{*}, Kaicheng Chen, Evangelos Tsotsas

Thermal Process Engineering, Otto von Guericke University, Universitätsplatz 2, 39106 Magdeburg, Germany

HIGHLIGHTS

- Build machine learning models to predict rod-like particle mixing in rotary drums.
- The training, validation, and test data are sourced from 121 DEM simulations.
- Models can predict mixing time and degree under four distinct independent variables.
- Each model has its pros and cons, but the overall predictions are satisfactory.
- Compared to DEM simulations, machine learning models save a lot of time.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:

Particle mixing
Rotary drum
DEM simulation
Non-spherical
Machine learning

ABSTRACT

The mixing of non-spherical particles in rotary drums exhibits significant complexity, particularly when density segregation and size segregation occur simultaneously. Three machine learning models: artificial neural network (ANN), extremely randomized trees (ERT), and particle swarm optimized support vector regression (PSO-SVR) were developed to predict the mixing time and mixing degree at the steady mixing state of rod-like particles in rotary drums. The training, validation, and test data for the machine learning models were generated from 121 discrete element method (DEM) simulations with four independent variables: revolution frequency, particle density ratio, particle size ratio, and drum length. All three models predicted the mixing degree accurately with $R^2 \geq 0.94$. The ERT and PSO-SVR models also predicted the mixing time well with $R^2 \geq 0.88$. Building machine learning models is hundreds of times faster than running DEM simulations, making these models highly promising for predicting larger-scale simulations with more complex-shaped particles.

^{*} Corresponding author.

E-mail address: wencong.wu@ovgu.de (W. Wu).

<https://doi.org/10.1016/j.powtec.2024.120307>

Received 29 July 2024; Received in revised form 9 September 2024; Accepted 21 September 2024

Available online 26 September 2024

0032-5910/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

1. Introduction

Particle mixing is quite common in various industrial fields, such as chemical, food and pharmaceutical industries. Rotary drums are widely popular as mixers due to their simple structure. Since the discrete element method (DEM) was proposed by Cundall and Strack in 1979 [1], it has gradually evolved into a powerful tool for analyzing particle flow within rotary drums [2–7]. In DEM, each particle is treated as an individual entity, and its motion is calculated based on Newton's second law. The DEM simulation can provide detailed insights into particle-level interactions and behaviors, which are difficult to achieve through traditional experiments.

However, the majority of past studies involving DEM have focused on spherical particles. One reason is that DEM was initially developed for spherical particles, as modeling spherical particles is much easier compared to non-spherical ones [8–11]. On the other hand, as a computationally intensive method, DEM has always been limited by the processing power of computers [12–14]. Not to mention, non-spherical particles require more computational resources compared to spherical ones, making it rather difficult to conduct large-scale simulations [15].

It is gratifying that in recent years, with the continuous deepening of research and the improvement of computer performance, there has been a gradual increase in studies using DEM to simulate the mixing behavior of non-spherical particles in drums. For example, He et al. [16] investigated size segregation of ellipsoids in rotary drums by DEM with number of particles varied from 10,000 to 56,000. The effects of aspect ratio, size ratio, and rotating speed on the extent of segregation were taken into consideration. Jiang et al. [17] used DEM to simulate the mixing of 22,000 spherical and cylindrical particles in a rotary drum. They identified the role of various parameters such as rotation speed, aspect ratio, density ratio as well as mass fraction and volume fraction of cylindrical particles. Dong et al. [18] established a super-quadric DEM model to describe the solid transportation mechanisms of 10,000 cylindrical particles in a rotary drum. Their study concerned the influence of aspect ratio and rotation speed on the granular flow behavior and mixing performance. Unfortunately, most of the related studies have only conducted analyses by varying one single independent variable at a time. The problem of predicting mixing conditions of non-spherical particles in rotary drums when multiple factors vary simultaneously remains largely unexplored. The reason is understandable, as this task is indeed challenging. When mixing non-spherical particles in rotary drums, they exhibit all the complexities inherent in the flow of spherical particles, including six different flow regimes [19–21], size and density segregation [22–24], radial streaks in the transverse plane [25,26] and alternating bands in the axial direction [27,28]. Additionally, there are different particle orientations and interlocking phenomena occur [18,29,30]. Such complex dynamics of non-spherical particles make the mixing outcome exhibit highly nonlinear relationships with different influencing factors. Conventional analysis methods struggle to accurately predict the results.

Thanks to the development of artificial intelligence, machine learning methods are increasingly being applied in engineering fields. They offer significant advantages in handling complex nonlinear problems, some of which involve particle mixing processes. For example, Miao et al. [31] developed a long-short term memory model to predict and classify particle segregation in rotary drums, their predictive results are consistent with DEM simulations; Dai et al. [32] applied the random forest algorithm for image processing, which accurately count the number of particles in a rotary drum on the radial direction; van Sleetuwen et al. [33] adopted the genetic optimization algorithm to estimate globally optimal particle parameters in scaling, dramatically decreasing the computational expense in DEM simulation. These contributions demonstrate the power of machine learning methods across various domains, including regression, classification, image processing, and parameter optimization. Machine learning methods not only surpasses traditional methods in terms of accuracy and speed but also addresses

key gaps in previous research, such as predicting particle mixing in rotary drums under simultaneous influence of multiple factors. Our previous research also established a DEM data-driven particle swarm optimized support vector regression model to predict spherical particle mixing in rotary drums, and the results of mixing time and mixing degree from this machine learning model were promising [34]. In the present work, a typical kind of non-spherical particles: rod-like particles, has been selected for DEM simulations. The structure of such particles is still relatively simple, and they are commonly used in industrial production. The rod-like particles are constructed with a multi-sphere model. Four independent variables are considered in the mixing simulations, which include revolution frequency of the drum, particle density ratio, particle size ratio, and drum length. The mixing results are quantitatively described using mixing time and mixing degree at the steady mixing state with the help of the subdomain-based mixing index (SMI). The differences in mixing results between spherical particles and rod-like particles are compared. Three machine learning models are developed for mixing prediction: artificial neural network (ANN), extremely randomized trees (ERT), and particle swarm optimized support vector regression (PSO-SVR). These three methods differ fundamentally in their modeling approaches. We extensively compared the differences among the three methods, not only in terms of their predictive accuracy, but also considering the implementation difficulty, interpretability, total modeling time, etc., hoping to provide reference and assistance to researchers who are not familiar with machine learning methods but plan to adopt them as predictive tools in their research. After establishing suitable machine learning models, the mixing behavior of non-spherical particles across the entire parameter space can be predicted accurately and rapidly. This overcomes a major challenge faced by traditional regression methods, which have difficulty accurately predicting non-spherical particle mixing when multiple factors change simultaneously. Additionally, the machine learning models operate hundreds of times faster than DEM simulations, highlighting their significant potential for predicting larger-scale simulations involving more complex particle shapes in industrial applications.

2. DEM simulation and mixing quantification

2.1. Multi-sphere model and simulation conditions

The superquadric model and the multi-sphere model are the two most popular models used to represent non-spherical particles in DEM simulations. The former utilizes the superquadric equation to define particles, constructing various non-spherical shapes through three shape parameters and two blockiness parameters. The latter combines primary spheres with the same or different sizes to an independent rigid body that represents a single non-spherical particle, overlapping between primary spheres being permitted. It is worth mentioning that although both models require significantly more computation time for simulating non-spherical particles compared to spherical ones, the reasons behind this are different. The time consumption of the superquadric model is primarily attributed to its complex contact detection algorithm [10]. On the other hand, the multi-sphere model employs the widely validated contact detection algorithm for spherical particles, where computational efficiency mainly depends on the number of primary particles [35]. Therefore, for simulating relatively simple non-spherical structures, using the multi-sphere model is a more efficient choice.

Our test mixing simulations in a rotary drum also confirm this point: under identical conditions, the CPU time for rod-like multi-sphere particles composed of four primary spheres is less than half of that for superquadric rod-like particles, and their mixing results are quite similar. Detailed settings and results of these test simulations are shown in [Appendix A](#).

For the above reason, in this study, all of the rod-like particles are structured by the multi-sphere model with four identical primary particles, resulting in an aspect ratio of 2:1, as shown in [Fig. 1](#).

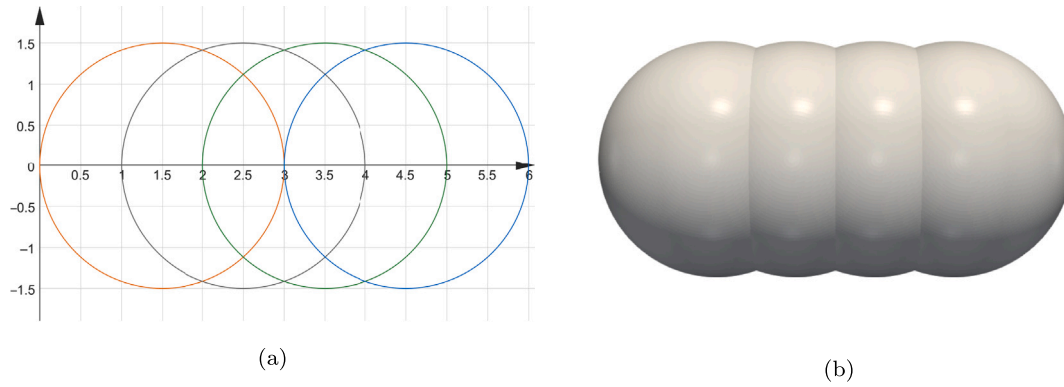


Fig. 1. Rod-like particle generated by the multi-sphere model: (a) coordinates of the primary particles; (b) combined rod-like particle.

Table 1

Key parameters and independent variables in the DEM simulations.

Key parameter	Value
Young's modulus [Pa]	1×10^7
Poisson's ratio [-]	0.24
Coefficient of restitution [-]	0.82
Friction coefficient [-]	0.80
Rolling coefficient [-]	0.01
Time step [s]	2×10^{-5}
Drum diameter [mm]	194
Filling degree [-]	0.2
Diameter of primary particle composing benchmark particle [mm]	3
Density of benchmark particle [kg/m^3]	1,000
Independent variable	Value
Revolution frequency [rpm]	10, 12, 15, 20, 25, 30, 40, 48, 50
Particle density ratio [-]	1, 1.2, 1.25, 1.3, 1.4, 1.5, 1.6, 1.7, 1.75, 1.8, 1.9, 2
Particle size ratio [-]	1, 1.2, 1.25, 1.5, 1.5, 1.75, 1.8, 2
Drum length [mm]	75, 100, 150, 200, 225, 300

We used the open-source software LIGGGHTS to conduct DEM simulations [36], which has a built-in multi-sphere model. At each time step, the total force and torque on each rigid multi-sphere particle is computed as the sum of the forces and torques on its constituent primary particles. The coordinates, velocities, and orientations of the primary particles in each rigid body are updated so that the multi-sphere particle moves and rotates as a single entity. Details of the governing equations as well as the contact model used in LIGGGHTS can be found in our previous paper [37].

The mixing simulations with binary mixtures of rod-like particles were performed in horizontal rotary drums. Before rotating, the two kinds of rod-like particles were generated in the drum with a top-bottom stratification. The particles initially generated and let fall to the bottom of the drum (so-called benchmark particles) always had constant size and density. The diameter of the primary spherical particles composing the rod-like benchmark particles is 3 mm, and the density of the benchmark particle is $1,000 \text{ kg}/\text{m}^3$. The particles latter generated and placed at the upper part of the bed had varied sizes and densities, with ratios relative to the benchmark particles. The size ratio of these two kinds of rod-like particles is equal to the diameter ratio of their primary particles. Key parameters as well as the independent variables that used in the DEM simulations are listed in Table 1, values of those key parameters have been validated through experiments for spherical particles in our previous work [37]. The range of drum revolution frequency (10–50 rpm) ensures that the flow regime of particle bed in the drum stays in rolling or cascading. These two regimes provide good particle movement, are commonly chosen for mixing operation in practice. The ranges of particle density ratio and size ratio (1–2) are set to be consistent with our previous work on simulating

spherical particles, facilitating comparison in subsequent sections. As for the range of drum length (75–300 mm), we take the length of the previous experimental drum, 150 mm, as an intermediate value and scale up and down by a factor of two. Regarding the specific values of the independent variables, the bold numbers in Table 1 represent those used in the training set of our machine learning models, while the regular numbers are used in the validation and test sets. More detailed data generation and grouping will be thoroughly explained in Section 3.1 later.

The total number of rod-like particles in the DEM simulations depends on the drum length and the particle size ratio, ranging from 5288 to 37,600. All the simulations were performed on a commercial computer with AMD Ryzen Threadripper 2950X processor (16 cores/32 threads, 3.5 GHz) and 32 GB RAM. A typical CPU time for a mixing process involving 18,800 rod-like particles (75,200 spherical primary particles) is 8.5 h. By comparison, under identical conditions, the CPU time for 75,200 spherical particles is 5.2 h.

2.2. Mixing quantification

We selected the subdomain-based mixing index (SMI) [38] for the quantitative description of mixing in this study, which first divides the whole mixing system into subdomains, and then obtains local mixing information from all subdomains, finally integrating them to get the overall mixing index. In comparison to the classic Lacey index [39], SMI considers all particles in the mixing system, thus avoiding biases caused by sampling. Compared with the widely used entropy of mixing [40], SMI ensures, through a clever algorithm, that the range of the local mixing index in each subdomain remains between 0 and 1, even for particles with different initial proportions. Moreover, SMI is not only suitable for binary mixtures but can also be used for mixtures with more particle types, making it widely applicable. The detailed formulas for SMI are placed in Appendix B.

The total number of subdomains in this work is kept at 1,250 ($5 \times 5 \times 50$) for all the drums. The subdomains are created by setting a 5×5 equidistant square grid with its edges tangent to the circular cross section of the drum and dividing the drum to 50 slices of equal thickness in the axial direction, as illustrated in Fig. 2. Using the same number of subdomains for mixers of different sizes means we always consider 1/1,250 of the entire system as the smallest unit to determine the mixing quality. This also ensures that when we simultaneously scale the drum and particle sizes, the overall mixing degree of the system remains unchanged. For each rod-like particle, its center of gravity is used as a marker to determine whether the particle is in a certain subdomain.

After obtaining the evolution curve of SMI over time for a mixing process, determining the mixing time and mixing degree at steady mixing state is the next task. These two parameters are of great importance for a mixing process, as they determine the process efficiency and the

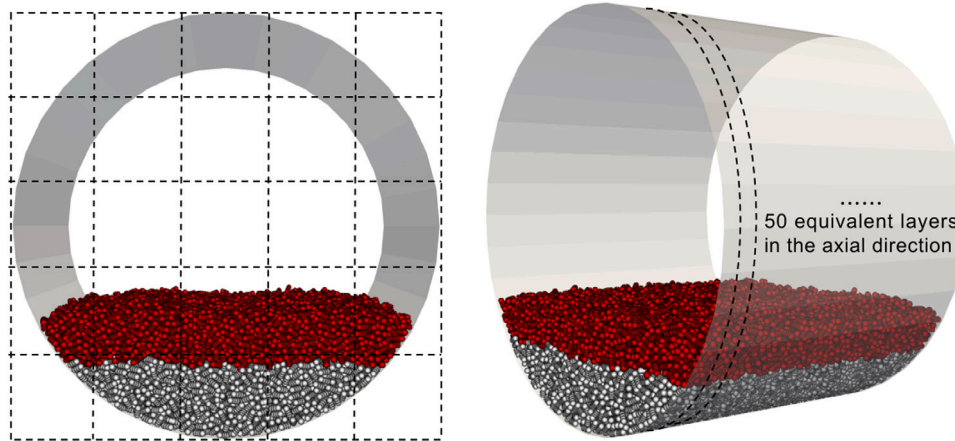


Fig. 2. Gridding manner for SMI.

product quality, respectively. When no segregation occurs during drum mixing, the shape of the SMI curve is quite consistent. The mixing degree continuously increases in the initial stage and stabilizes after reaching a steady mixing state, with the values fluctuating around a constant. Such curves can be conveniently fitted using exponential functions. However, this method is not suitable for the mixing of particles with different densities and sizes. Because when segregation phenomena occur, the value of mixing degree may decrease over time, indicating that the final mixing degree at steady mixing state may be lower than a top value achieved during the mixing process. In order to handle the different types of mixing curves, we process the SMI evolution as follows, keeping it consistent with our previous research on spherical particles [34]:

- (1) Run each simulation for 60 s and check the difference in SMI between 50 s and 60 s. If the difference is less than 0.02, then average the SMI values over 50–60 s as the mixing degree at steady mixing state, otherwise continue the simulation for further 60 s and repeat the same procedure until steady state SMI has been found;
- (2) Calculate the moving average of the original SMI curve as the smoothed SMI curve by the `smooth` function in MATLAB with a span of the moving average of 49 considered data points to reduce noise or fluctuations;
- (3) Lay two horizontal straight lines from right to left on the image of the smoothed SMI curve, the values of which should correspond to 95% and 105% of the mixing degree at steady state, respectively. Find the first intersection point between the curve and the lines, and take the corresponding time as the mixing time.

Fig. 3 gives examples of the above mentioned determination process for two typical shapes of mixing curves.

2.3. Mixing of rod-like particles vs. mixing of spherical particles

In this section, the simulation results of rod-like particles mixing will be compared with the results of spherical particles from our previous studies. We start with the simplest scenario, setting both the density ratio and the size ratio to unity. No segregation occurs in this scenario. The drum length here is 150 mm. The results regarding mixing times and mixing degrees at steady mixing state under different revolution frequencies are shown in Fig. 4. It is evident that the mixing of rod-shaped particles is faster than that of spherical particles, up to reaching the maximum drum revolution frequency (50 rpm) in our simulations, where the mixing times of both types of particles become relatively close. This may be because individual rod-like particles experience more collisions than spherical particles, and

due to the interlocking phenomenon of rod-like particles, sliding is significantly less compared to spherical particles, thereby increasing the efficiency of mixing. Furthermore, increasing the revolution frequency has a more gradual impact on the mixing time of spherical particles, with their relationship approaching linearity. However, for rod-like particles, increasing the revolution frequency beyond 20 rpm does not significantly affect the mixing time. As for the mixing degree at steady mixing state, mixing of spherical particles is slightly more uniform compared to rod-like particles. Steady SMI values are not significantly affected by the revolution frequency in either case.

In the next scenario, the size ratio is still set to unity, but the density ratio is changed. Only density segregation appears in this scenario. The revolution frequency and the drum length here are also kept constant at 10 rpm and 150 mm, respectively. The influences of the density ratio on mixing can be found in Fig. 5. For both rod-like particles and spherical particles, the occurrence of density segregation dramatically decreases the mixing time to quite similar values. However, regarding the mixing degree at steady mixing state, the difference between the two shapes of particles is obvious. Starting from the density ratio of 1.25, the steady SMI of rod-like particles is always higher than that of spherical ones, indicating that the effect of density segregation is more pronounced for spherical particles.

A deeper understanding can be gained by comparing the mixing curves for the two shapes of particles as well as their mixing patterns at steady mixing state. The cases with a density ratio equal to two serve as an example, as shown in Fig. 6. The trends of the two mixing curves are similar, but the curve for the rod-like particles fluctuates more violently. For spherical particles at steady mixing state, heavy particles (blue) are almost completely surrounded by the light particles (white), resulting in a relatively low degree of mixing. While observing the mixing of rod-like particles at steady state, although there are also more light particles (white) on the surface of the particle bed, they are still far from enclosing the heavy particles (red). Moreover, the dynamic angle of repose of rod-like particles is evidently greater than that of spherical particles. Rod-like particles also exhibit more jagged edges of the particle bed. All these features indicate that the movement of rod-like particles is more intense than that of spherical particles, which in turn leads to a higher degree of mixing.

The last scenario is similar to the previous one, but changes in the size ratio are analyzed. In this scenario, only size segregation occurs. Specifically, the revolution frequency is 30 rpm, the density ratio is set to unity, and the drum length is equal to 150 mm. Here 30 rpm is chosen instead of 10 rpm because the mixing time of spherical particles at low revolution frequency is very long, which is not convenient for comparison. The effects of the size ratio on mixing for rod-like particles and spherical particles are shown in Fig. 7. The mixing time and mixing degree at steady mixing state for rod-like particles and

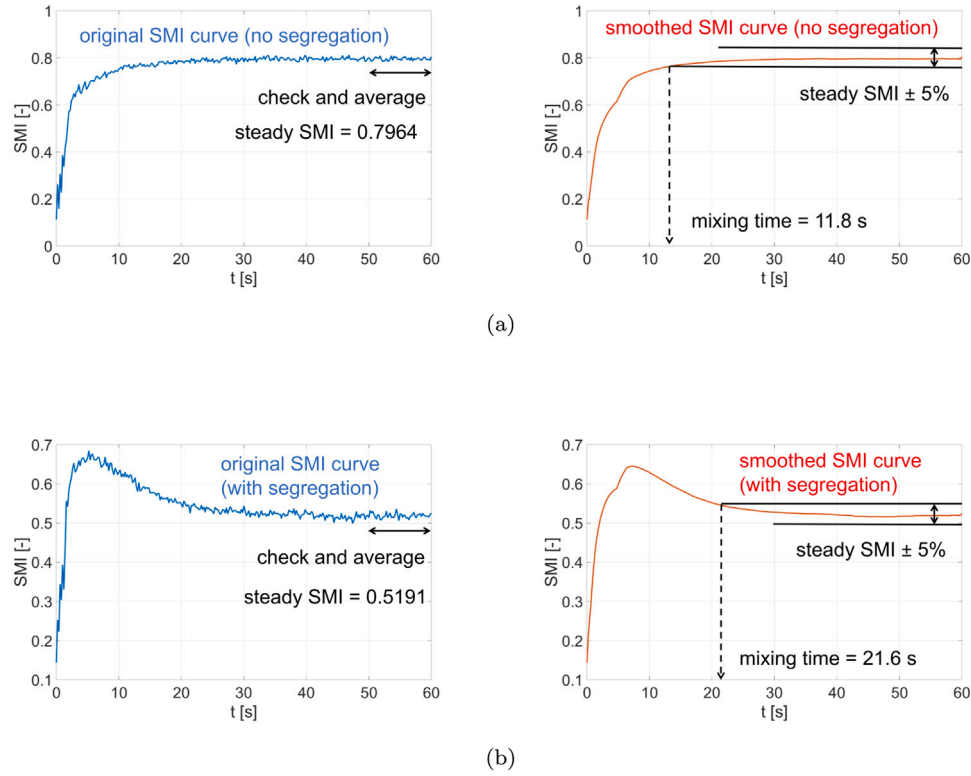


Fig. 3. Examples of determining the mixing degree at steady mixing state and the mixing time for different SMI evolution: (a) without segregation (revolution frequency = 50 rpm, density ratio = 1, size ratio = 1, drum length = 150 mm); (b) with density and size segregation (revolution frequency = 50 rpm, density ratio = 1.5, size ratio = 1.5, drum length = 150 mm).

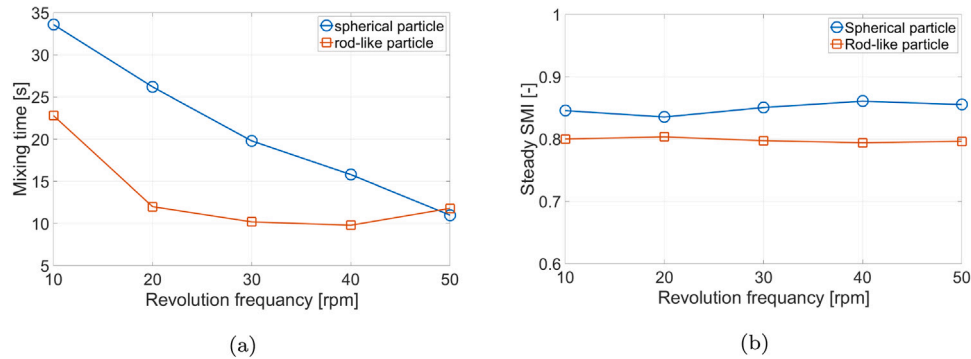


Fig. 4. Compare mixing between rod-like particles and spherical particles without segregation phenomenon (density ratio = 1, size ratio = 1, drum length = 150 mm): (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

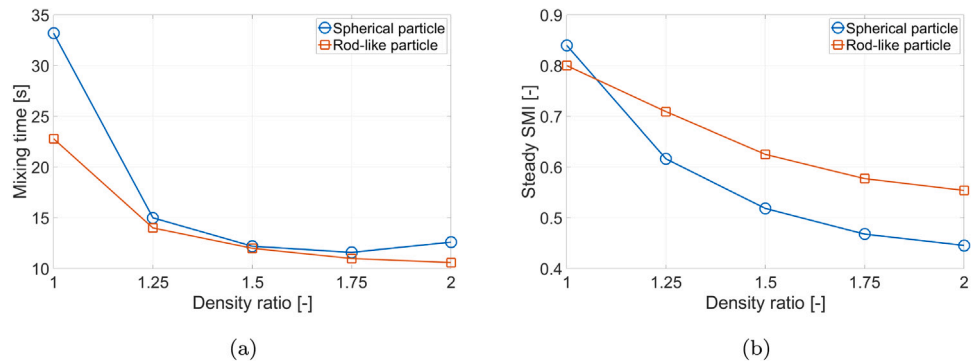


Fig. 5. Compare mixing between rod-like particles and spherical particles with only density segregation phenomenon (revolution frequency = 10 rpm, size ratio = 1, drum length = 150 mm): (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

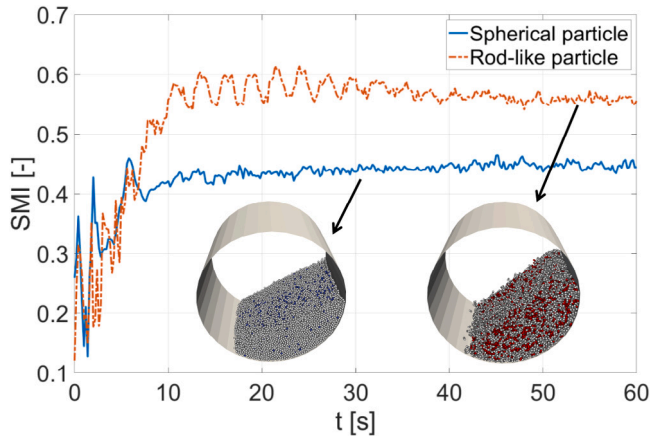


Fig. 6. Mixing curves and mixing patterns at steady mixing state for rod-like particles and spherical particles with density ratio of two (white particles are the benchmark particles with relatively low density).

spherical particles exhibit significant differences. When size segregation occurs with spherical particles, the mixing times dramatically increase, reaching a maximum at the size ratio of 1.5. Conversely, for rod-like particles, their mixing times become slightly shorter after size segregation appears. As for the mixing degree at steady mixing state, the steady SMI of both rod-like particles and spherical particles decreases. However, the difference is that for spherical particles, their decreasing trend gradually levels out, with a size ratio of two appearing to be the limit; further increasing the size ratio does not result in poorer mixing. In contrast, the decrease in steady SMI of rod-like particles is nearly linear, and its value may continue decreasing even after the size ratio exceeds two.

More detailed information for this scenario is also obtained by checking the mixing curves and observing the mixing patterns at steady mixing states, as shown in Fig. 8. Here, both rod-like particles and spherical particles have a size ratio of two. The shapes of the two mixing curves are completely different. The mixing degree of rod-like particles increases sharply and approaches the steady state rapidly. But for the spherical particles, after a short period in which the mixing degree increases, the SMI drops slowly until it reaches a steady level, leading to a prolonged mixing time and poor mixing quality. The mixing pattern for the spherical particles clearly shows that three distinct alternating particle bands in the axial direction have been caused by size segregation: large particles (blue) accumulate near the ends of the drum, while small particles (white) are located in the middle of the particle bed. For rod-like particles, although there is a higher concentration of large particles (red) near the ends of the drum compared to the center, clear alternating bands are not formed, resulting in significantly higher mixing quality than in case of spherical particles.

In the above analysis, we compared the mixing of rod-like particles and the mixing of spherical particles for relatively simple mixing scenarios: no segregation, only density segregation, and only size segregation. For rod-shaped particles, although the mixing time and mixing degree at steady mixing state differ among those scenarios, it seems that they can all be described by simple regression models. However, when both segregation phenomena occur simultaneously and multiple independent variables change concurrently, the mixing situation of rod-like particles becomes extremely complex. The changes in mixing time and degree exhibit highly nonlinear behavior, as shown in Fig. 9. Many counter-intuitive results emerged, for example use of a longer drum not necessarily prolonged the mixing time and increase of the revolution frequency not always improved the mixing quality. Analyzing such complex relationships effectively is virtually beyond reach, let alone making predictions. In this kind of situation, accurate predictions of

mixing time and mixing degree at steady mixing state for rod-like particles may be best achieved through artificial intelligence-based machine learning methods.

3. Machine learning models for prediction

3.1. Data generation and grouping

In order to make predictions using machine learning models, we took the results from 121 DEM simulations of mixing rod-like particles in rotary drums. Four independent variables varied in the simulations (revolution frequency of the drum, particle density ratio, particle size ratio, and drum length) are taken as the input parameters in the machine learning models. The two variables that quantitatively describe the mixing state: mixing time and mixing degree (SMI) at steady mixing state, are used as output parameters in the machine learning models. These 121 simulations are grouped into three sets: a training set (81), a validation set (20), and a test set (20). The training set is used to train and develop the model. The validation set is used for hyperparameter tuning or adjusting the model during training. Its specific role for different models will be explained in detail in the following sections. The test set is used to evaluate the final performance of the fully established model using entirely new data compared to the training and validation sets, and it does not influence the model itself.

In the training set, each input parameter has three levels, resulting in 81 sets of simulations through full factorial design, as shown in Table 1 before. This kind of design guarantees a comprehensive and effective training. The number of simulations in the test set was determined based on a classic Pareto principle (also known as the 80/20 rule) in machine learning practice, i.e., amount of training data : amount of test data = 8 : 2. The number of simulations in the validation set remained the same as in the test set. The input parameters of validation and test set were chosen within the parametric range established by the training set, but with different values. Using a fixed data split ensures consistency and fairness when comparing machine learning methods across different platforms, as it avoids the variability introduced by random data partitioning in cross-validation. This fixed split approach also facilitates the reproducibility of model results.

Additionally, it should be noted that after determining the machine learning model, for prediction of the two output parameters in this study, two separate models will be established. Hyperparameter settings for those two models may differ. There are three reasons for this procedure: most machine learning models do not support multiple outputs due to algorithmic limitations; even for models that support multiple outputs, such as artificial neural networks (ANN), the accuracy of two single-output models is much higher than that of a single dual-output model; the two output parameters in this study are not highly correlated, so there is no need to compute both parameters simultaneously. Nevertheless, the data grouping for the two output parameters is identical.

3.2. Artificial neural network model

Artificial neural networks (ANN) are popular machine learning models which have been inspired by the biological neural networks of the human brain [41]. The neurons in the human brain receive electrical signals from other neurons, and when the potential of a neuron exceeds a certain threshold, it becomes activated and sends signals to the next neuron. The mathematical model of neurons in ANN is very similar to this process. A typical artificial neuron with n inputs from the previous neurons or the original input data is shown in Fig. 10. Each input is weighted with an appropriate w and then summed up. The bias b in the artificial neuron is like the threshold in the biological neuron. The sum of the weighted inputs, along with the bias, is transformed by the activation function to form the output of this neuron. The role of the activation function is quite crucial: on the one

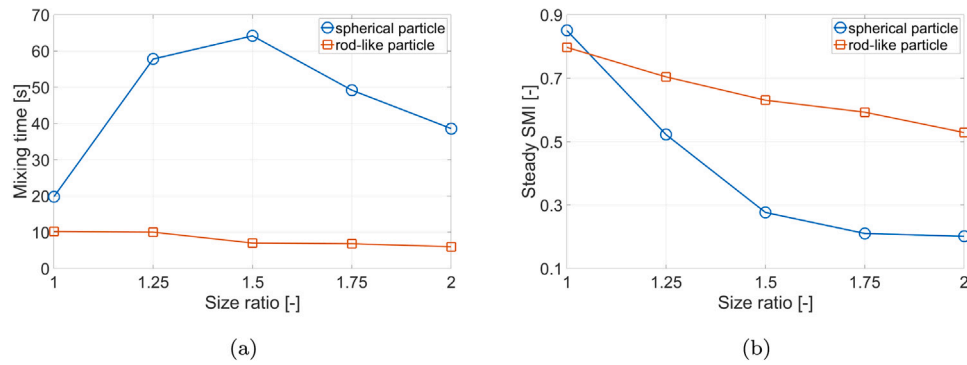


Fig. 7. Compare mixing between rod-like particles and spherical particles with only size segregation phenomenon (revolution frequency = 30 rpm, density ratio = 1, drum length = 150 mm): (a) mixing times; (b) mixing degrees (SMI) at steady mixing state.

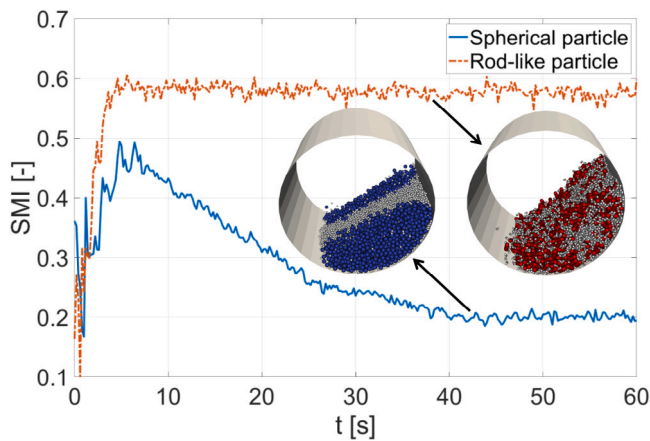


Fig. 8. Mixing curves and mixing patterns at steady mixing state for rod-like particles and spherical particles with size ratio of two (white particles are the benchmark particles with relatively small size).

hand it squeezes the input values, which may vary over a wide range, into a specific range of output values; and, on the other hand, it allows for a nonlinear correlation to be established between the input and output values. Without the activation function, no matter how complex the neural network structure is, it is impossible to predict a nonlinear relationship. A commonly used activation function – the tan-sigmoid transfer function – is presented as an example in Fig. 10, the output values from which is in the range of $(-1, 1)$.

Feedforward neural networks are the most common network structure for regression modeling. Specifically, each layer of neurons is fully interconnected with the next layer of neurons, and there are no same-layer connections between neurons or cross-layer connections. Fig. 11 gives an example of a simple feedforward ANN that consists of one hidden layer with three neurons and one output layer with one neuron. The network can increase in complexity by increasing the number of hidden layers as well as the number of neurons in the layers. By observing the network structure, we can also understand why the ANN is able to implement multiple outputs very easily. For additional outputs, it is only necessary to add the appropriate number of neurons in the output layer and build connections with neurons in the last hidden layer, without modifying the rest of the network structure.

The ANN models in this study were implemented by the deep learning toolbox in MATLAB. In both predictive models for mixing time and mixing degree at steady mixing state, we used the tan-sigmoid transfer function `tansig` as activation function in the hidden layer and the linear transfer function `purelin` as activation function in the output layer. The `purelin` function is an identity mapping function that does not impose any restrictions on the range of output values

and is commonly used in the output layer for regression problems. The number of neurons in each hidden layer was kept as the default value ten. For predicting the steady SMI, we chose the default single hidden layer network structure, while for predicting the mixing time, we increased the number of hidden layers to two. This is because our trial runs revealed that a single hidden layer network led to insufficient training when predicting mixing time, resulting in low accuracy on the training set. However, increasing the number of hidden layers to three caused significant overfitting, where the model performed extremely well on the training set but poorly on the test set. This further indicates that predicting mixing time is indeed more challenging than predicting mixing degree at steady mixing state, as the range of variation in mixing time is much larger than that of mixing degree. Training an ANN for regression is to tune the values of the weights and biases of the network to optimize the predictive performance. The performance function for our networks during training is the mean square error, and the training algorithm here is Levenberg–Marquardt backpropagation [42,43]. This algorithm combines the advantages of gradient descent and Newton's method and is generally the fastest algorithm in the deep learning toolbox. Since this algorithm is also based on gradient optimization, differences in the scales of input parameters will lead to unstable and slow gradient updates for multi-input models, hence data preprocessing is necessary. The function `mapminmax` was applied to normalize input parameters to fall in the range $[-1, 1]$.

As mentioned before, the validation set plays different roles in different machine learning models. During each epoch of training the ANN model, the performance of the model on the validation set was monitored synchronously to decide when the training process should be stopped, which is known as the early stopping technique. With this technique, overfitting of the ANN model can be effectively avoided. In our ANN model, the training process finished when the validation error was larger than or equal to the previously smallest validation error for six consecutive validation iterations. Taking the ANN model which predicts mixing degree at steady mixing state as an example, the mean squared errors of different data sets during training are shown in Fig. 12. The training process stopped at epoch 14 because at epoch 8 the best validation performance has been reached. It is evident that the error in the training set continuously decreases with the increase in training epochs, which is not the case for the validation set and the test set. If the epoch keeps increasing, the model will be at risk of overfitting. Moreover, the test set error and the validation set error show similar characteristics, which indicates that the data grouping of these two sets is reasonable.

It is worth noting that due to the random initialization of weights and biases in the network, even when constructing the network with exactly the same hyperparameters, the output may have slight differences and cannot be entirely reproduced. The prediction results of mixing time and mixing degree at steady mixing state in the test set are shown in Fig. 13. The ANN model does not perform very well in predicting

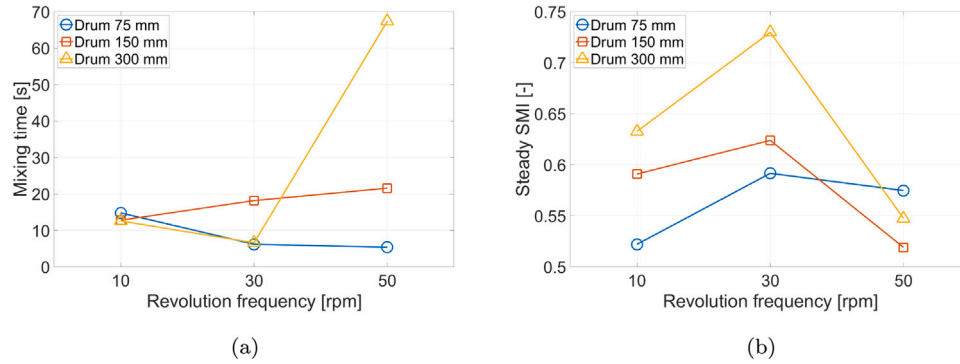


Fig. 9. Relationship between mixedness and revolution frequency for rod-like particles in drums of different lengths with both density and size segregation (density ratio = 1.5, size ratio = 1.5): (a) mixing time (b) mixing degree (SMI) at steady mixing state.

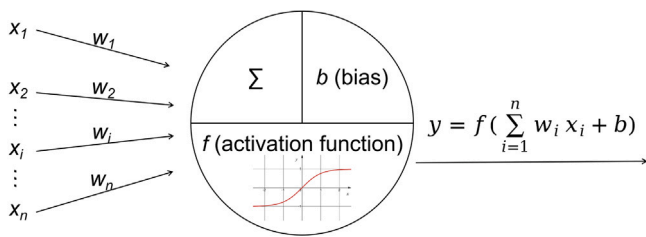


Fig. 10. A typical artificial neuron.

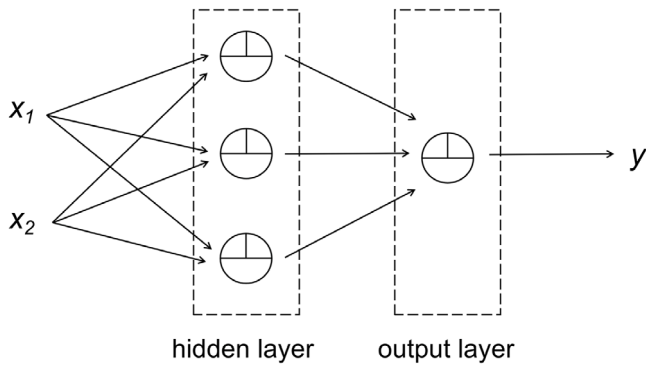


Fig. 11. The structure of a simple artificial neural network.

mixing time, with a coefficient of determination (R^2) of only 0.76. Most of the mixing times are underestimated, and the mean absolute error (MAE) is 5.1 s. This result is mainly due to the inherent difficulty in predicting mixing time, as the maximum value (67.4 s) and the minimum value (5 s) in the training set differ by more than 13 times. Another possible reason is that ANN performs better with large-scale data, and its performance on highly nonlinear, small-scale data cannot be guaranteed. On the contrary, ANN shows really high accuracy ($R^2 = 0.95$) in predicting the mixing degree at steady mixing state. Moreover, the data points are evenly distributed on both sides of the reference line, indicating that the model here has no systematic bias. Its MAE is also quite low, only 0.02.

3.3. Extremely randomized trees model

Extremely randomized trees (ERT) is a tree-based ensemble learning method introduced by Geurts et al. in 2006 [44]. As an improvement and extension of the traditional decision tree approach, ERT not only introduce randomness during the tree-growing process to speed up

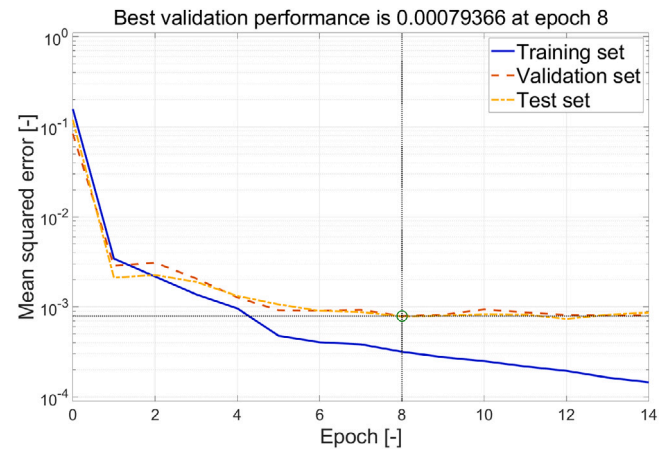


Fig. 12. Training performance of the ANN model predicting mixing degree at steady mixing state.

the calculation, but also aggregate the results from multiple trees to enhance the robustness and accuracy of the model.

A simple example is used to illustrate how a single decision tree, the basic unit of ERT, handles a regression problem, as shown in Fig. 14. The data in this example have two input parameters x_1, x_2 and one output parameter y . The decision tree utilizes the binary tree structure. At each growth step, one of the input parameters and a threshold value will be selected as a node to split the data (ellipse text box), and the input space will also be divided into two sub-regions. After growing, the output value for each sub-region corresponds to the mean output of all data points contained within that sub-region.

The improvement of ERT over the traditional regression tree lies mainly in the selection of candidate nodes. A traditional regression tree traverses through all the possible values of each input parameter. An ERT, on the other hand, will randomly select only one threshold within the range of minimum and maximum values of each input parameter. Subsequently, for all candidate nodes, the criterion for selecting the optimal one is to minimize the mean squared error of the split data. Because ERT has far fewer candidate split points compared to the traditional regression tree, its computational efficiency is considerably higher. During the training process of ERT, multiple independent decision trees are constructed using the method described above. When making predictions, the output of ERT is the average of results from all the individual decision trees.

In this study, the ERT model has been implemented through an open source machine learning library in Python named Scikit-learn [45]. One notable advantage of ERT over other machine learning methods is the simplicity of its hyperparameter settings, especially for small-scale

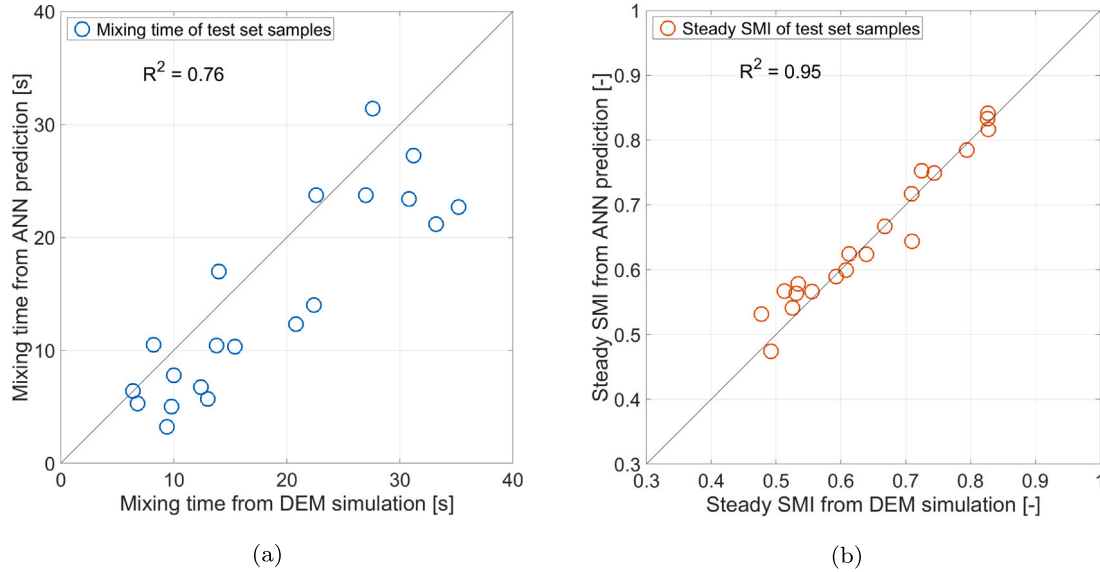


Fig. 13. Predictive accuracy of ANN model on the test set: (a) mixing time; (b) mixing degree (SMI) at steady mixing state.

data with few input parameters. Because our data only has four input parameters, we set `max_features=None`, which means ERT always considers all these four input parameters during splitting instead of a random subset. Also because our training set only has 81 samples, we can let the trees fully develop. In other words, no limitations are imposed on the depth of the tree growth and a node can continue to split as long as it contains at least two samples (`max_depth=None` and `min_samples_split=2`). The number of trees in ERT is set as 10, instead of the default value 100. This decision was based on the results of our trial runs, which showed that increasing the number of trees from 10 to 100 had minimal impact on the prediction accuracy, but increased the modeling time by a factor of 7.6. Consequently, this hyperparameter is determined as 10, which not only mitigates the potential overfitting that can occur with single decision trees but also ensures efficient computation. Additionally, because the ERT use data splitting for training and each input parameter has its own independent dimension in the input space, there is no need to perform any normalization or standardization preprocessing on the training data. This also simplifies the modeling process compared to other machine learning methods.

Even after determining all hyperparameters, the results of ERT can vary significantly with each run. This variation is because the threshold for each split in ERT is randomly chosen. However, the ERT implemented in Python allows for fixing the random splitting of trees using the `random_state` parameter. This ensures consistent results across different runs. The task of the validation set here is to find the optimal random state. Specifically, we use the training set to build models under different random states, compare the predictive accuracy of these models on the validation set, select the random state with the highest R^2 as a fixed parameter for the model, and finally test the model performance on new data from the test set. We tried 2,000 different random states from `random_state=0` to `random_state=1999`, and found that the optimal random state for mixing time was `random_state=1188`, whereas for the mixing degree at steady mixing state it was `random_state=329`. The results of their predictions on the test set are shown in Fig. 15. The accuracy of ERT for predicting mixing time is acceptable with R^2 of 0.88. Apart from a few data points with relatively large deviations, the rest of the data points are evenly distributed on both sides of the reference line, with a MAE of only 2.4 s. This demonstrates that ERT, being insensitive to the scale of input parameters, has a clear advantage when handling data with significant differences in the upper and lower limits. The predictive performance

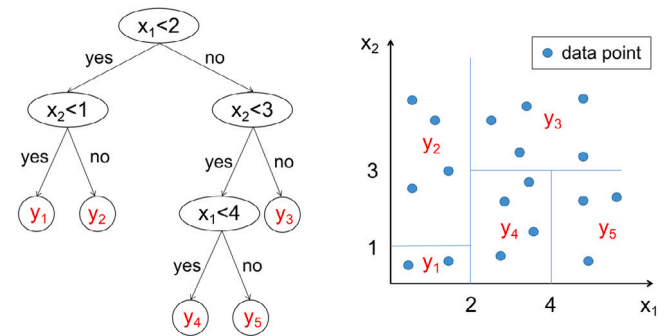


Fig. 14. A decision tree for regression: structure of the tree and division of the input space.

of ERT for mixing degree at steady mixing state is even better, with R^2 achieving a value of 0.94. No systematic bias is observed. And, the MAE here is only 0.02.

Due to the intuitive and transparent modeling process of ERT, it has stronger interpretability compared to most black-box machine learning algorithms. This is particularly evident in the calculation of feature importance. In the field of machine learning, the input parameters of a model are also named as features. Feature importance analysis can quantify the impact of each feature on the output results. Since the predicted mean square errors before and after the split of candidate nodes are calculated each time during the growth of trees in ERT, the results can be naturally used as a measure of the feature importance. The greater the change in the mean square error before and after the split, the more important the feature is. This change is accumulated according to each feature, normalized, and then averaged over the results of all trees to obtain the final feature importance. The results of feature importance on mixing time and mixing degree at steady mixing state are shown in Fig. 16. The influences of the four input parameters on the mixing time vary greatly. But for the mixing degree at steady mixing state, except for the revolution frequency, the other three input parameters have relatively even effects. For both output parameters, the drum revolution frequency has the smallest influence, while the particle size ratio has the largest impact. This indicates that, in order to improve the mixing efficiency and the mixing quality of the rod-like particles, it is better to avoid size segregation caused by different particle sizes rather than increasing the revolution frequency.

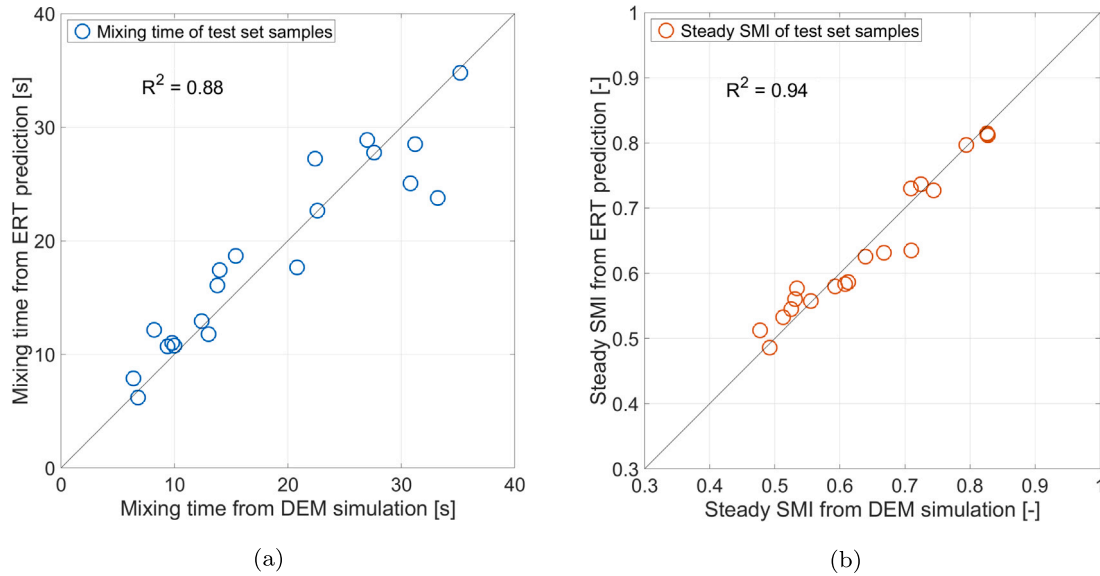


Fig. 15. Predictive accuracy of ERT model on the test set: (a) mixing time; (b) mixing degree (SMI) at steady mixing state.

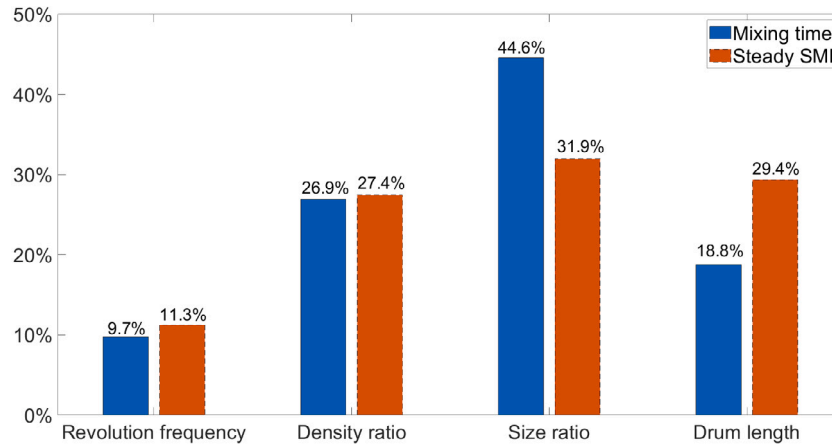


Fig. 16. Feature importance of mixing time and mixing degree (SMI) at steady mixing state from ERT model.

3.4. Particle swarm optimized support vector regression model

Support vector machines are classic machine learning algorithms used for classification tasks [46]. Support vector regression (SVR) extends the principle of support vector machines to regression problems, where the objective is to predict continuous outcomes rather than discrete class labels [47]. This model has significant advantages when dealing with high-dimensional data with non-linear relationships, and it also performs well for small-scale data. The mathematical principles behind SVR are much more complex than those of the previously mentioned methods (ANN and ERT). This section aims to introduce the core principles of the SVR algorithm in a simple and intuitive manner. More detailed formulas can be found in our previous research [34] or other Refs. [48,49].

The basic idea behind SVR can be better understood by comparing it with linear regression, as shown in Fig. 17. Although one-dimensional data (only one input parameter) is used as an example in the figure, the formulas use a more general vector notation to indicate that the model is also applicable to multi-dimensional data. Notably, the derivation of regression equations $f(x)$ is the common goal of both models, linear regression and SVR, the only difference lies in their objective functions. In linear regression, the objective is to minimize the sum of the squared errors between the predicted and actual values, considering all data

points. However, in SVR, an ϵ -insensitive tube is created. The tube is designed to contain as many data points as possible, with the straight line (or hyperplane for multi-dimensional data) in the center of the tube representing the regression equation. The distance from the edge to the center of the tube in the y -direction, ϵ , also known as the margin of tolerance, is the first hyperparameter in SVR, which needs to be set before training. Data points located within this tube are considered to be perfectly predicted and do not contribute to the loss calculation. Losses are calculated only for data points that fall on the edge of or outside the tube, with the values of the differences between the actual values and the predicted values exceeding ϵ . The objective function of SVR consists of two terms. The first term is the regularization term of the model, which controls the complexity of the model; the second term refers to the loss function of the training error. The penalty coefficient C in the second term is the second hyperparameter in SVR, which balances the predictive ability of the model on the training data with the generalization ability of the model for new data.

While dealing with non-linear relationships, the strategy of SVR is to introduce the kernel trick, which implicitly maps the input data to a high-dimensional feature space, transforming the original non-linear problem to a linear problem for solution. The radial basis function kernel was chosen in this work since it provides a good flexibility while

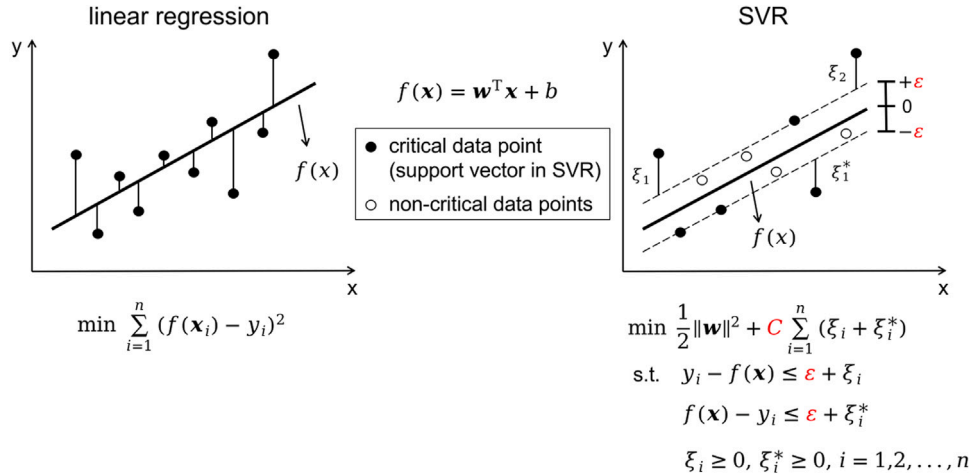


Fig. 17. Comparison between linear regression and SVR for one dimensional data.

ensuring calculation speed. The expression of this kernel is:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right). \quad (1)$$

where γ is the kernel function coefficient, serving as the third hyperparameter in our SVR model.

The solution process of SVR does not contain any randomness, so the final prediction model is unique for each hyperparameter combination. In other words, setting hyperparameters appropriately is crucial for the SVR model to achieve good predictive performance. There are three hyperparameters: margin of tolerance ϵ , penalty coefficient C and kernel function coefficient γ in the SVR model, and none of them is an integer, which makes the hyperparameter tuning really challenging. Hence, a metaheuristic optimization algorithm, namely particle swarm optimization (PSO), was chosen to address this challenge. This algorithm is inspired by the social behavior of birds within a flock [50]. The particles in PSO are not physical but represent potential solutions of the optimization problem. The PSO efficiently explores the entire search space to find the global optimal solution, making it highly suitable for hyperparameter tuning in SVR.

The objective of PSO in this work is to find the best combination of hyperparameters (ϵ, C, γ) which gives the highest predictive accuracy (R^2) in the validation set. A group of particles, also called the swarm, is generated with random initialized positions, i.e., values of (ϵ, C, γ) at the beginning of the algorithm. In each iteration, particles move one step, termed velocity in PSO, within the search space, guided by their personal best position and the global best position of the swarm, gradually converging to the optimal solution, which can be expressed as [51]:

$$\mathbf{v}_{i,r+1} = w\mathbf{v}_{i,r} + c_1 r_1 (\mathbf{p}_{i,r}^{best} - \mathbf{p}_{i,r}) + c_2 r_2 (\mathbf{g}_r^{best} - \mathbf{p}_{i,r}), \quad (2)$$

$$\mathbf{p}_{i,r+1} = \mathbf{p}_{i,r} + \mathbf{v}_{i,r+1}, \quad (3)$$

where $\mathbf{p}_{i,r} = (C, \epsilon, \gamma)$ is the position vector of particle i in iteration r , and $\mathbf{v}_{i,r}$ is the corresponding velocity vector. The personal best position $\mathbf{p}_{i,r}^{best}$ signifies the best (ϵ, C, γ) combination found by particle i up to iteration r . The global best position \mathbf{g}_r^{best} indicates the best combination discovered by the entire swarm up to iteration r . In these expressions, w represents the inertia weight controlling the influence of previous velocity on the current velocity, while c_1 and c_2 are the acceleration coefficients regulating the impact of personal best and global best positions on the particle's velocity. Additionally, r_1 and r_2 are random numbers sampled from a uniform distribution between zero and one. To balance exploration and exploitation of the search space, the inertia weight w is set to 0.4, and the acceleration coefficients are both set to $c_1 = c_2 = 2$. For balancing predictive accuracy and

computational efficiency, the particle number in the swarm and the maximum iteration are both set to 20.

The particle swarm optimized support vector regression (PSO-SVR) model was implemented in Python, utilizing Scikit-learn for the SVR part and a self-programmed PSO part. All input parameters were standardized using the StandardScaler function. Through PSO optimization, the optimal hyperparameter combinations for predicting mixing time and mixing degree at steady state were determined as $(\epsilon, C, \gamma) = (1.98, 89.4, 0.37)$ and $(0.001, 16.5, 0.076)$, respectively. The margin of tolerance ϵ in SVR for predicting mixing time is notably larger than that for predicting mixing degree, reflecting broader variations and higher noise levels in the mixing time data, hence requiring a wider tube, consistent with previous observations. The performances of PSO-SVR models on the test set are shown in Fig. 18. The PSO-SVR model provides good accuracy ($R^2 = 0.90$) when predicting mixing time. The MAE is only 2.4 s with generally no systematic bias. The performance of PSO-SVR model on predicting steady SMI is excellent with $R^2 = 0.96$. The MAE of mixing degree is quite low, with the value of 0.02. The data points closely align with the reference line.

3.5. Model comparison and discussion

Three machine learning models (ANN, ERT, and PSO-SVR) have been developed to predict mixing time and mixing degree at steady mixing state for rod-like particles in rotary drums based on DEM simulations, each with distinct mathematical modeling principles. All 121 sets of DEM simulation data and the predictions from the three machine learning models are summarized in Appendix C. In this section, these three models will be comprehensively compared, including their predictive performances and various characteristics, to aid researchers using similar machine learning models in their studies. While much of the content has been covered in previous sections, our focus here is to organize and summarize these findings, as presented in Table 2.

The prediction performance of the three models for the new data on the test set is first summarized, which is the most important criterion for measuring the reliability of a machine learning model, and their training set results are also given below as a reference. All three models perform well when predicting the mixing degree at steady mixing state in the test set, and their differences are quite small. However, for predicting mixing time, the ANN is significantly less accurate than the other two models. As mentioned earlier, the predictive accuracy of the ANN model cannot be guaranteed on small-scale data. A deeper reason may lie in a critical hyperparameter of ANN: the number of hidden layers, which can only change in discrete integers. This constraint has a significant impact on the model training process on small-scale data. In our trial runs, we found that with one hidden layer, the ANN model

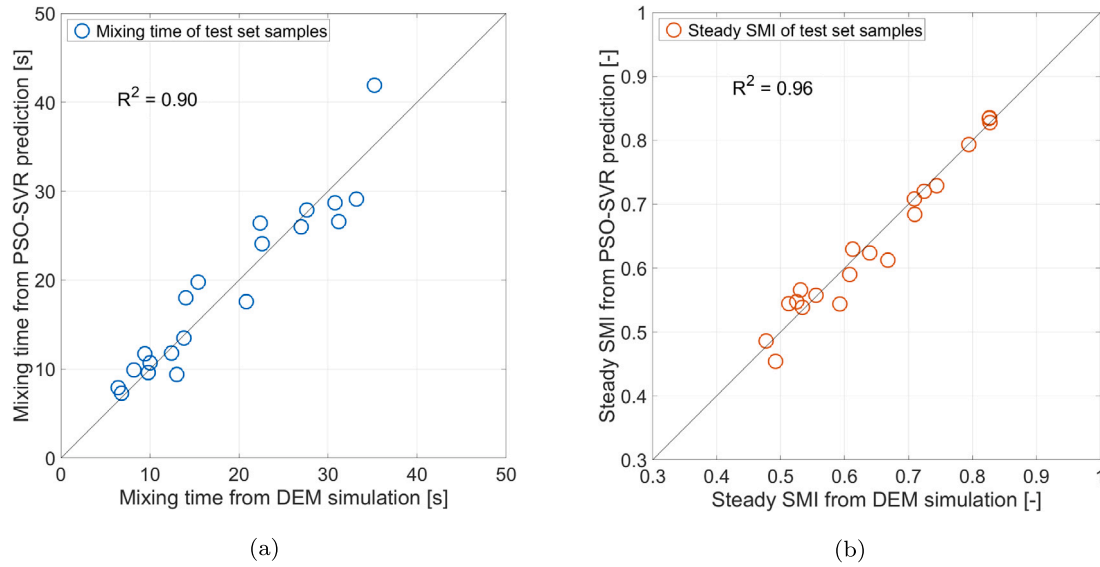


Fig. 18. Predictive accuracy of PSO-SVR model on the test set: (a) mixing time; (b) mixing degree (SMI) at steady mixing state.

Table 2

Comparison of three machine learning methods.

	ANN	ERT	PSO-SVR
Test set			
R ² of mixing time	0.76	0.88	0.90
MAE of mixing time [s]	5.1	2.4	2.4
R ² of steady SMI	0.95	0.94	0.96
MAE of steady SMI [-]	0.021	0.022	0.018
Training set			
R ² of mixing time	0.92	1	0.94
MAE of mixing time [s]	2.0	0	2.1
R ² of steady SMI	0.97	1	0.98
MAE of steady SMI [-]	0.014	0	0.008
Software platform	Matlab	Python	Python
Difficulty of implementation	Easy	Moderate	Hard
Reproducibility	No	Yes	Yes
Interpretability	Weak	Strong	Moderate
Data preprocessing	Necessary	Unnecessary	Necessary
Role of validation set	Early stopping	Determining random state	Hyperparameter tuning
Total modeling time [s]	8.4	27.1	100.8

was severely undertrained when predicting mixing time, while with three hidden layers, the model exhibited severe overfitting. Finally, the number of hidden layers was set to two. However, even with this adjustment, the R² of the ANN model on the training set was still the lowest among the three models, reaching only 0.92, indicating that the model training may still be insufficient. This could probably be improved by adding more training data. Another notable aspect of the training set results is the performance of the ERT model. The predictions from this method were absolutely identical to the original data, which is a direct consequence of the ERT's training algorithm. Although our ERT model consists of 10 trees, with their node thresholds randomly selected, each tree was fully grown. This means that each tree completely partitions the 81 data points in the training set according to different divisions of the input space, perfectly matching the input-output pairs. As a result, even after averaging, this mapping remains unchanged.

In terms of model implementation, the deep learning toolbox in MATLAB provides a very comprehensive graphical interface for ANN, making model development relatively simple and clear. Implementing ERT with the help of Scikit-learn requires basic Python syntax, while using PSO for hyperparameter tuning in SVR requires more advanced Python programming skills. Since the weights and biases in ANN are

randomly initialized, exact reproduction of the model results is impossible, unlike the other two models written in Python. The ERT model has the strongest interpretability among the three models since the calculation of feature importance is embedded in its modeling process. The PSO-SVR model has moderate interpretability because the values of some hyperparameters can partially reflect the data characteristics. In terms of handling the data, ERT is insensitive to the scale of the features, so preprocessing is not necessary. The last row of the table shows the total modeling time, encompassing the entire training, validation, and testing processes of the model. Since each machine learning model makes separate predictions for mixing time and mixing degree at steady mixing state, and the time for each run is slightly different, we employ an averaging and summing approach to calculate the total modeling time. Specifically, for a certain machine learning model, we predict the mixing time and steady SMI five times each, and then average their modeling times separately, finally add these two averages as its total modeling time. The PSO-SVR model has the longest total modeling time (100.8 s), primarily due to the time spent on hyperparameter tuning (validation) through PSO. Nevertheless, compared to DEM simulations, which take hours to run a single case, a machine learning model is a significant time saver. This advantage is further extended when

performing simulations with a larger scale and more complex particle shapes.

In practical industrial applications, the obtained raw data is unlikely to be as comprehensive as the training data generated through the full factorial design in this study. In such cases, cross-validation can be employed to divide the training and validation sets, and with a sufficient amount of data, similar training effectiveness can still be achieved, without affecting the potential of machine learning methods. Returning to the problem of non-spherical particle mixing in rotary drums, this study has successfully predicted the mixing time and mixing degree at steady mixing state of rod-like particles under simultaneous variation of multiple key factors. However, the particle mixing process in rotary drums is quite complex, and several other influencing factors have not yet been considered, such as filling degree of the drum, ternary or higher-order mixtures, and difficult particle shapes, all of which can be explored in future research. Enhancing the interpretability of machine learning models is also a valuable direction for future efforts.

4. Conclusion

We performed 121 discrete element method (DEM) simulations of rod-like particles mixed in rotary drums. The rod-like particles were constructed using a multi-sphere model. Four independent variables were varied in the simulations: revolution frequency of the drum, particle density ratio, particle size ratio, and drum length. Two important characteristics in the mixing process, mixing time and mixing degree at steady mixing state, were defined with the help of the subdomain-based mixing index (SMI). Three simple mixing scenarios: no segregation, only density segregation, and only size segregation were compared between rod-like particles and spherical particles. Due to the high complexity of mixing results when multiple independent variables are simultaneously varied, predicting these outcomes becomes very challenging. Therefore, we developed three machine learning prediction models: artificial neural network (ANN), extremely randomized trees (ERT), and particle swarm optimized support vector regression (PSO-SVR). All three models provide excellent performance ($R^2 \geq 0.94$) in predicting mixing degree at steady mixing state. The predictive accuracy of ERT ($R^2 = 0.88$) and PSO-SVR ($R^2 = 0.90$) for mixing time is also good, better than that of ANN ($R^2 = 0.76$). The strengths and weaknesses of the three models were compared in detail. In general, machine learning models not only provide high accuracy in predicting complex relationships but also offer significant time-saving advantages over DEM simulations, making them highly promising for rapid predictions in larger-scale particle processing equipment with more complex particle shapes.

CRediT authorship contribution statement

Wencong Wu: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **Kaicheng Chen:** Writing – review & editing, Supervision, Project administration. **Evangelos Tsotsas:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Attached in Supplementary Material.

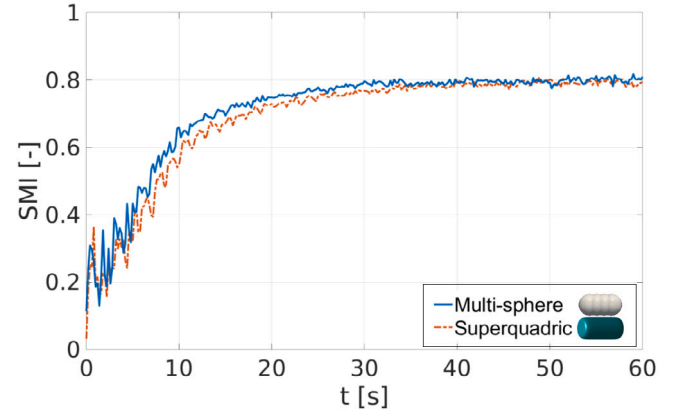


Fig. A.1. Mixing curves of the test simulations, in which rod-like particles are created either by the multi-sphere model or by the superquadric model.

Acknowledgments

The authors would like to gratefully acknowledge the funding of this work by the German Federal Ministry of Education and Research (BMBF) within the project ADMIX (03INT609BA).

Appendix A. Compare multi-sphere model and superquadric model

The test simulations were performed in a 150 mm long rotary drum with 10 rpm, the density ratio and size ratio were set to 1. For both non-spherical particle models, the total number of rod-like particles is 18,800. The combination method of the multi-sphere model has been described in the main text, as shown in Fig. 1. In the superquadric model, the surface of a superquadric particle can be described by the inside–outside function given by Barr [52]:

$$\left(\left|\frac{x}{a}\right|^{n_2} + \left|\frac{y}{b}\right|^{n_2}\right)^{n_1/n_2} + \left|\frac{z}{c}\right|^{n_1} - 1 = 0. \quad (\text{A.1})$$

Here, the three shape parameters are set as $a = b = 1.5$ mm, $c = 3$ mm, and the blockiness parameters are $n_1 = 8$, $n_2 = 2$. In both DEM simulations 60 s of mixing process have been simulated, and the CPU times of the multi-sphere model and the superquadric model were 8.5 h and 18.3 h, respectively. The mixing curves for both models are shown in Fig. A.1.

Appendix B. Formulas of subdomain mixing index (SMI)

The subdomain mixing index (SMI) is applicable for mixtures of more than two particle types with different initial proportions. Assume a mixture with Q types of particles in a mixing system that is divided into n subdomains. The participation factor of particle type k can be calculated as:

$$f_k = \frac{\max(\sum_{i=1}^n N_{1i}, \sum_{i=1}^n N_{2i}, \dots, \sum_{i=1}^n N_{Qi})}{\sum_{i=1}^n N_{ki}}. \quad (\text{B.1})$$

Here N_{ki} is the number of type k particles in the subdomain i . The modified fraction of k -type particles in subdomain i can be computed as:

$$P_{ki} = \frac{N_{ki} f_k}{\max(N_{1i} f_1, N_{2i} f_2, \dots, N_{Qi} f_Q)}. \quad (\text{B.2})$$

The local SMI for subdomain i is:

$$\text{SMI}(i) = \frac{1}{Q-1} \left(\sum_{k=1}^Q P_{ki} - 1 \right). \quad (\text{B.3})$$

Finally, the total mixing index is obtained by averaging all the local SMIs:

$$SMI = \frac{1}{N_{tot}} \sum_{i=1}^n \left[SMI(i) \sum_{k=1}^Q N_{ki} \right], \quad (B.4)$$

where N_{tot} is the total number of particles in the whole mixing system. The range of SMI is between zero and one: zero indicates a completely separated state, whereas one means that different types of particles are perfectly mixed.

Appendix C. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.powtec.2024.120307>.

References

- [1] P.A. Cundall, O.D. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1) (1979) 47–65.
- [2] M. Kwapinska, G. Saage, E. Tsotsas, Mixing of particles in rotary drums: A comparison of discrete element simulations with experimental results and penetration models for thermal processes, *Powder Technol.* 161 (1) (2006) 69–78.
- [3] L. Zhang, Z. Jiang, F. Weigler, F. Herz, J. Mellmann, E. Tsotsas, PTV measurement and DEM simulation of the particle motion in a flighted rotating drum, *Powder Technol.* 363 (2020) 23–37.
- [4] L. Zhang, Z. Jiang, J. Mellmann, F. Weigler, F. Herz, A. Bück, E. Tsotsas, Influence of the number of flights on the dilute phase ratio in flighted rotating drums by PTV measurements and DEM simulations, *Particuology* 56 (2021) 171–182.
- [5] X. Tang, Y. Yue, S. Wang, Y. Shen, Modelling of gas-solid-liquid flow and particle mixing in a rotary drum, *Powder Technol.* 409 (2022) 117758.
- [6] X. Tang, S. Wang, X. Jin, Y. Shen, Super-quadric CFD-DEM modelling of chip-like particle-liquid flow in a rotary drum, *Powder Technol.* 435 (2024) 119363.
- [7] Q. Jian, H. Gu, K. Wang, S. Wang, M. Zhan, J. Wang, L. Ji, Z. Chi, G. Zhang, Numerical study of particle behaviours and heat transfer in a complex rotary kiln, *Particuology* 92 (2024) 81–94.
- [8] A. Podlozhnyuk, S. Pirker, C. Kloss, Efficient implementation of superquadric particles in Discrete Element Method within an open-source framework, *Comput. Part. Mech.* 4 (2017) 101–118.
- [9] B. Soltanbeigi, A. Podlozhnyuk, S.-A. Papanicopolulos, C. Kloss, S. Pirker, J.Y. Ooi, DEM study of mechanical characteristics of multi-spherical and superquadric granular particles at micro and macro scales, *Powder Technol.* 329 (2018) 288–303.
- [10] Y. You, Y. Zhao, Discrete element modelling of ellipsoidal particles using super-ellipsoids and multi-spheres: A comparative study, *Powder Technol.* 331 (2018) 179–191.
- [11] D. Peng, K.J. Hanley, Contact detection between convex polyhedra and superquadrics in discrete element codes, *Powder Technol.* 356 (2019) 11–20.
- [12] H. Kruggel-Emden, M. Sturm, S. Wirtz, V. Scherer, Selection of an appropriate time integration scheme for the Discrete Element Method (DEM), *Comput. Chem. Eng.* 32 (10) (2008) 2263–2279.
- [13] J. Xu, H. Qi, X. Fang, L. Lu, W. Ge, X. Wang, M. Xu, F. Chen, X. He, J. Li, Quasi-real-time simulation of rotating drum using Discrete Element Method with parallel GPU computing, *Particuology* 9 (4) (2011) 446–450.
- [14] Y. He, T. Evans, A. Yu, R. Yang, A GPU-based DEM for modelling large scale powder compaction with wide size distributions, *Powder Technol.* 333 (2018) 219–228.
- [15] B. Yan, R.A. Regueiro, A comprehensive study of MPI parallelism in three-dimensional Discrete Element Method (DEM) simulation of complex-shaped granular particles, *Comput. Part. Mech.* 5 (4) (2018) 553–577.
- [16] S. He, J. Gan, D. Pinson, A. Yu, Z. Zhou, Radial segregation of binary-sized ellipsoids in a rotating drum, *Powder Technol.* 357 (2019) 322–330.
- [17] C. Jiang, X. An, M. Li, Y. Wu, D. Gou, Y. Wu, DEM modelling and analysis of the mixing characteristics of sphere-cylinder granular mixture in a rotating drum, *Powder Technol.* 426 (2023) 118653.
- [18] T. Dong, S. Yang, S. Wang, Super-quadric DEM study of cylindrical particle behaviours in a rotating drum, *Powder Technol.* 437 (2024) 119511.
- [19] J. Mellmann, The transverse motion of solids in rotating cylinders: Forms of motion and transition behavior, *Powder Technol.* 118 (3) (2001) 251–270.
- [20] R.Y. Yang, A. Yu, L. McElroy, J. Bao, Numerical simulation of particle dynamics in different flow regimes in a rotating drum, *Powder Technol.* 188 (2) (2008) 170–177.
- [21] I. Govender, Granular flows in rotating drums: A rheological perspective, *Miner. Eng.* 92 (2016) 168–175.
- [22] D. Khakhar, A.V. Orpe, S. Hajra, Segregation of granular materials in rotating cylinders, *Phys. A* 318 (1–2) (2003) 129–136.
- [23] E. Alizadeh, O. Dubé, F. Bertrand, J. Chaouki, Characterization of mixing and size segregation in a rotating drum by a particle tracking method, *AIChE J.* 59 (6) (2013) 1894–1905.
- [24] S. Yang, Y. Sun, L. Zhang, J.W. Chew, Segregation dynamics of a binary-size mixture in a three-dimensional rotating drum, *Chem. Eng. Sci.* 172 (2017) 652–666.
- [25] N. Jain, J.M. Ottino, R.M. Lueptow, Regimes of segregation and mixing in combined size and density granular systems: An experimental study, *Granul. Matter* 7 (2) (2005) 69–81.
- [26] G. Pereira, S. Pucilowski, K. Liffman, P. Cleary, Streak patterns in binary granular media in a rotating drum, *Appl. Math. Model.* 35 (4) (2011) 1638–1646.
- [27] K. Hill, A. Caprihan, J. Kakalios, Axial segregation of granular media rotated in a drum mixer: Pattern evolution, *Phys. Rev. E* 56 (4) (1997) 4386.
- [28] H. Kuo, R. Hsu, Y. Hsiao, Investigation of axial segregation in a rotating drum, *Powder Technol.* 153 (3) (2005) 196–203.
- [29] H. Ma, Y. Zhao, Modelling of the flow of ellipsoidal particles in a horizontal rotating drum based on DEM simulation, *Chem. Eng. Sci.* 172 (2017) 636–651.
- [30] M. Li, Y. Wu, Y. Qian, X. An, H. Li, DEM simulation on mixing characteristics and macroscopic/microscopic flow behaviors of different-shaped spherocylinders in a rotating drum, *Ind. Eng. Chem. Res.* 60 (24) (2021) 8874–8887.
- [31] Q. Miao, W. Zhu, J. Wang, P. Huang, C. Hu, Y. Ge, Application of the combination on data-driven and discrete element model: Particle segregation prediction and classification in rotary drums, *Chem. Eng. Res. Des.* 200 (2023) 456–468.
- [32] J. Dai, H. Xu, Y. Gao, J. Yu, X. Gao, Measuring spherical and nonspherical binary particles: Mixing and segregation in a rotating drum using machine learning-assisted image processing, *Ind. Eng. Chem. Res.* 63 (17) (2024) 7891–7901.
- [33] R. van Sleetuwen, S. Pantaleev, M. Ebrahimi, et al., Efficient DEM modeling of solid flavor particle mixing in a rotary drum, *Powder Technol.* 437 (2024) 119559.
- [34] W. Wu, K. Chen, E. Tsotsas, Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model, *Powder Technol.* 434 (2024) 119365.
- [35] Z. Jiang, J. Du, C. Rieck, A. Bück, E. Tsotsas, PTV experiments and DEM simulations of the coefficient of restitution for irregular particles impacting on horizontal substrates, *Powder Technol.* 360 (2020) 352–365.
- [36] C. Kloss, C. Goniva, A. Hager, S. Amberger, S. Pirker, Models, algorithms and validation for opensource DEM and CFD-DEM, *Prog. Comput. Fluid Dynam. Int. J.* 12 (2–3) (2012) 140–152.
- [37] W. Wu, K. Chen, E. Tsotsas, Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation, *Adv. Powder Technol.* 33 (4) (2022) 103512.
- [38] M. Cho, P. Dutta, J. Shim, A non-sampling mixing index for multicomponent mixtures, *Powder Technol.* 319 (2017) 434–444.
- [39] P.M.C. Lacey, Developments in the theory of particle mixing, *J. Appl. Chem.* 4 (5) (1954) 257–268.
- [40] M. Schutyser, J. Padding, F. Weber, W. Briels, A. Rinzeema, R. Boom, Discrete particle simulations predicting mixing behavior of solid substrate particles in a rotating drum fermenter, *Biotechnol. Bioeng.* 75 (6) (2001) 666–675.
- [41] W.S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.* 5 (1943) 115–133.
- [42] K. Levenberg, A method for the solution of certain non-linear problems in least squares, *Quart. Appl. Math.* 2 (2) (1944) 164–168.
- [43] D.W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *J. Soc. Ind. Appl. Math.* 11 (2) (1963) 431–441.
- [44] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, *Mach. Learn.* 63 (2006) 3–42.
- [45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-Learn: Machine learning in Python, *J. Mach. Learn. Res.* 12 (2011) 2825–2830.
- [46] C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.* 20 (1995) 273–297.
- [47] H. Drucker, C.J. Burges, L. Kaufman, A. Smola, V. Vapnik, Support vector regression machines, *Adv. Neural Inf. Process. Syst.* 9 (1996).
- [48] C.-C. Chang, C.-J. Lin, Libsvm: A library for support vector machines, *ACM Trans. Intell. Syst. Technol.* 2 (3) (2011) 1–27.
- [49] M. Awad, R. Khanna, *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*, Apress, Berkeley, CA, 2015, pp. 67–80, Ch. Support Vector Regression.
- [50] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *Proceedings of ICNN'95, International Conference on Neural Networks*, vol. 4, IEEE, 1995, pp. 1942–1948.
- [51] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360)*, IEEE, 1998, pp. 69–73.
- [52] A.H. Barr, Superquadrics and angle-preserving transformations, *IEEE Comput. Graph. Appl.* 1 (01) (1981) 11–23.

Chapter 5

Summary and Outlook

5.1 Summary

Solid particle mixing is a fundamental industrial unit operation with a long history and widespread applications. Mixing time and mixing degree are two of the most critical aspects of a mixing process, one is closely related to process efficiency and the other determines product quality. However, even in geometrically simple systems like horizontal rotary drums, accurately predicting these two aspects remains a challenge. This difficulty arises not only from the relatively underdeveloped theoretical framework of solid dynamics compared to fluid mechanics but also from the multitude of factors influencing particle mixing. A particularly noteworthy phenomenon in particle mixing is segregation, which occurs primarily due to differences in the nature of the solid particles involved.

Discrete element method (DEM) has proven to be a powerful simulation tool for studying granular mixing, with its effectiveness validated by numerous experimental studies. By combining DEM simulations with a mixing index, which is able to quantitatively describe a mixing process, both the mixing time and the mixing degree can be accurately predicted. However, the computational cost of DEM remains a significant limitation, particularly in applications requiring frequent parameter adjustments, such as automated industrial processes. While existing studies mainly focus on the qualitative influence of individual variables on mixing, the core objective of this research is to rapidly and accurately predict particle mixing in rotary drums under varying influencing factors based on a limited number of DEM simulations.

This cumulative dissertation consists of a comprehensive introduction and three published manuscripts. These manuscripts propose two types of approaches for predicting particle mixing in rotary drums, both relying on DEM simulation data and consistently employing the subdomain-based mixing index (SMI) as the sole criterion for determining mixing time and mixing degree. In Chapter 1, the DEM models for spherical and non-spherical particles are introduced in detail, including the numerical integration algorithm and representation of non-spherical particles' orientation—both of which were not covered in

the three manuscripts. This is followed by a discussion on the determination of key DEM parameters, including an explanation of the skin distance, another concept not addressed in the manuscripts. Furthermore, SMI is compared with two other commonly used mixing indices, and reasons for selecting SMI as the preferred metric are thoroughly examined. These additions significantly enrich the content of the three manuscripts and provide a clearer understanding of the research foundation and overall framework.

Chapter 2 proposes the first type of approach, termed cross-correlation, which enables scale-spanning correlation between particle mixing systems of different sizes. In this manuscript, the two systems considered were a 3D rotary drum and a 2D rotary drum, both containing spherical particles. The reliability of DEM simulations was validated through an experiment combined with image analysis, establishing a solid foundation for all subsequent studies. The mixing results of particles at different revolution frequencies in both the 3D and 2D drums were used as the training set for cross-correlation. The predictive performance of cross-correlation was evaluated both within and beyond the parametric range of the training set. Additionally, the applicability and limitations of cross-correlation were analyzed, and its advantages over machine learning methods were discussed.

Chapter 3 presents the second type of approach, which is a machine learning model called particle swarm optimized support vector regression (PSO-SVR). This model considers four independent variables: revolution frequency, density ratio, size ratio as well as drum length, and is able to predict the mixing time as well as the mixing degree at steady mixing state of spherical particles mixed in rotary drums. The model was trained using data from 81 DEM simulations, the input parameters of which were generated through a full factorial design with three levels for each of the four independent variables. The mixing results from the training set were subsequently analyzed. An overview of the SVR algorithm was provided, and PSO was employed to optimize the three hyperparameters of the SVR model. Finally, the predictive performance of the PSO-SVR model was evaluated by comparing its results on both the training and test sets with DEM simulations, and the outcomes were visualized using parity plots.

Chapter 4 extends the machine learning approach introduced in Chapter 3 from spherical particles to a common type of non-spherical particles—rod-like particles. Three different machine learning models, namely artificial neural networks (ANN), extremely randomized trees (ERT), and PSO-SVR, were employed to predict the mixing behavior of rod-like particles in rotary drums. In the DEM simulations, rod-like particles were represented using the multi-sphere model. The input and output variables of the three machine learning models in this manuscript remained consistent with those of the PSO-SVR model in Chapter 3. The DEM simulation results for rod-like and spherical particles were compared under three different scenarios: no segregation, only density segregation, and only size segregation, incorporating the observed mixing patterns within the drums. Each of the

three machine learning models was briefly introduced, followed by a comparison of the model predictions and DEM results in the test set using parity plots. A comprehensive evaluation was conducted at the end, summarizing various performance aspects in a table, such as predictive accuracy, difficulty of implementation, and total modeling time. This analysis aims to provide reference and assistance for researchers seeking to apply machine learning models in similar predictive tasks.

5.2 Outlook

It is important to note that while this study does not introduce new insights into the mechanisms of particle mixing and segregation in rotary drums, nor does it contribute to related dynamic analyses, these aspects remain the most fundamental and critical topics in the field of granular mixing. Any breakthrough in these areas would undoubtedly result in benefits for research related to the prediction of particle mixing.

The SMI was chosen as the sole mixing index in this study due to its numerous advantages; however, its full potential has not been fully explored. For instance, SMI is capable of quantitatively describing the mixing state of polydisperse mixtures with more than two components and can be used to analyze local mixing characteristics, such as the axial mixing distribution in long rotary drums. Nevertheless, as a fraction-based mixing index, SMI is inherently influenced by subdomain size. To completely eliminate this limitation, either a novel distance-based mixing index must be developed, or existing distance-based indices need to be optimized to accommodate large-particle systems. In this regard, the neighbor list in DEM presents a promising concept that may serve as a useful reference.

The transition phase from no segregation to the onset of segregation is particularly critical. Whether for studying the underlying mechanisms or improving predictive accuracy, it is essential to increase the number of simulations in this range. Specifically, more simulations should be conducted for size and density ratios between 1 and 1.5.

One major advantage of the cross-correlation method over machine learning models is its ability to make predictions beyond the parametric range of the training set, provided that the considered parameter does not significantly affect the axial motion of particles. Therefore, its applicability may be further explored, including scenarios such as varying filling degrees, mixing cohesive particles, or using flighted rotary drums. Moreover, the study by Xie et al. [119] introduced an interesting approach, suggesting that non-spherical particle behavior in DEM can be approximated by increasing the rolling resistance of spherical particles. This approach could also be integrated into the cross-correlation framework.

When applying machine learning methods in this study, spherical and non-spherical particles were investigated separately. However, particle shape could potentially be

abstracted as an additional influencing parameter, such as circularity. Saeed et al. [120] have already explored this approach in a cylindrical tank mixer. While more influencing factors can certainly be incorporated into the machine learning model, doing so would inevitably lead to an increased number of training samples. Therefore, it is necessary to explore alternative fractional factorial design methods beyond full factorial design. Finally, as most machine learning models function as black boxes, enhancing their interpretability remains a worthwhile direction for future research.

Bibliography

- [1] P. M. C. Lacey, Developments in the theory of particle mixing, *Journal of Applied Chemistry* 4 (1954) 257–268.
- [2] L. Zhang, Z. Jiang, J. Mellmann, F. Weigler, F. Herz, A. Bück, E. Tsotsas, Influence of the number of flights on the dilute phase ratio in flighted rotating drums by PTV measurements and DEM simulations, *Particuology* 56 (2021) 171–182.
- [3] C. Rößler, M. Riegler, Simulation in wood science: using the process gluing as an example, *Simulation Notes Europe* 30 (2020) 67–72.
- [4] M. Rhodes (Ed.), *Introduction to Particle Technology*, John Wiley & Sons, 2008.
- [5] H. G. Merkus, G. M. Meesters (Eds.), *Production, Handling and Characterization of particulate materials*, Springer, 2016.
- [6] P. Tang, V. Puri, Methods for minimizing segregation: A review, *Particulate Science and Technology* 22 (2004) 321–337.
- [7] H. Xiao, D. McDonald, Y. Fan, P. B. Umbanhowar, J. M. Ottino, R. M. Lueptow, Controlling granular segregation using modulated flow, *Powder Technology* 312 (2017) 360–368.
- [8] D. B. Hastie, P. W. Wypych, Segregation during gravity filling of storage bins, in: *IUTAM Symposium on Segregation in Granular Flows: Proceedings of the IUTAM Symposium held in Cape May, NJ, USA June 5–10, 1999*, Springer, 2000, pp. 61–72.
- [9] F. D. Cúñez, D. Patel, R. C. Glade, How particle shape affects granular segregation in industrial and geophysical flows, *Proceedings of the National Academy of Sciences* 121 (2024) e2307061121.
- [10] N. Jain, J. M. Ottino, R. M. Lueptow, Regimes of segregation and mixing in combined size and density granular systems: An experimental study, *Granular Matter* 7 (2005) 69–81.
- [11] H. Kuo, R. Hsu, Y. Hsiao, Investigation of axial segregation in a rotating drum, *Powder Technology* 153 (2005) 196–203.

- [12] J. Mosby, S. R. de Silva, G. G. Enstad, Segregation of particulate materials—mechanisms and testers, *KONA Powder and Particle Journal* 14 (1996) 31–43.
- [13] D. Zhang, Z. Zhou, D. Pinson, DEM simulation of particle stratification and segregation in stockpile formation, in: *EPJ Web of Conferences*, volume 140, EDP Sciences, 2017, p. 15018.
- [14] T. Zhang, J. Gan, D. Pinson, Z. Zhou, Size-induced segregation of granular materials during filling a conical hopper, *Powder Technology* 340 (2018) 331–343.
- [15] R. Hogg, D. Cahn, T. Healy, D. Fuerstenau, Diffusional mixing in an ideal system, *Chemical Engineering Science* 21 (1966) 1025–1038.
- [16] H. Henein, J. K. Brimacombe, A. P. Watkinson, Experimental study of transverse bed motion in rotary kilns, *Metallurgical Transactions B* 14 (1983) 191–205.
- [17] J. Mellmann, The transverse motion of solids in rotating cylinders—forms of motion and transition behavior, *Powder Technology* 118 (2001) 251–270.
- [18] R. Y. Yang, A. Yu, L. McElroy, J. Bao, Numerical simulation of particle dynamics in different flow regimes in a rotating drum, *Powder Technology* 188 (2008) 170–177.
- [19] N. Taberlet, P. Richard, J. Hinch, S shape of a granular pile in a rotating drum, *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 73 (2006) 050301.
- [20] J. Mellmann, E. Specht, X. Liu, Prediction of rolling bed motion in rotating cylinders, *AIChE Journal* 50 (2004) 2783–2793.
- [21] X. Y. Liu, E. Specht, O. G. Gonzalez, P. Walzel, Analytical solution for the rolling-mode granular motion in rotary kilns, *Chemical Engineering and Processing: Process Intensification* 45 (2006) 515–521.
- [22] H. Komossa, S. Wirtz, V. Scherer, F. Herz, E. Specht, Transversal bed motion in rotating drums using spherical particles: Comparison of experiments with DEM simulations, *Powder Technology* 264 (2014) 96–104.
- [23] H.-T. Chou, C.-F. Lee, Cross-sectional and axial flow characteristics of dry granular material in rotating drums, *Granular Matter* 11 (2009) 13–32.
- [24] Q. Zheng, A. Yu, Modelling the granular flow in a rotating drum by the Eulerian finite element method, *Powder Technology* 286 (2015) 361–370.
- [25] X. Xiao, Y. Tan, H. Zhang, R. Deng, S. Jiang, Experimental and DEM studies on the particle mixing performance in rotating drums: Effect of area ratio, *Powder Technology* 314 (2017) 182–194.

- [26] R. K. Soni, R. Mohanty, S. Mohanty, B. Mishra, Numerical analysis of mixing of particles in drum mixers using DEM, *Advanced Powder Technology* 27 (2016) 531–540.
- [27] J. Hlosta, L. Jezerská, J. Rozbroj, D. Žurovec, J. Nečas, J. Zegzulka, DEM investigation of the influence of particulate properties and operating conditions on the mixing process in rotary drums, Part 2: Process validation and experimental study, *Processes* 8 (2020) 184.
- [28] N. Nityanand, B. Manley, H. Henein, An analysis of radial segregation for different sized spherical solids in rotary cylinders, *Metallurgical Transactions B* 17 (1986) 247–257.
- [29] M. Arntz, H. Beeftink, W. K. den Otter, W. J. Briels, R. Boom, Segregation of granular particles by mass, radius, and density in a horizontal rotating drum, *AIChE Journal* 60 (2014) 50–59.
- [30] D. Khakhar, J. McCarthy, J. M. Ottino, Radial segregation of granular mixtures in rotating cylinders, *Physics of Fluids* 9 (1997) 3600–3614.
- [31] G. G. Pereira, M. D. Sinnott, P. Cleary, K. Liffman, G. Metcalfe, I. D. Šutalo, Insights from simulations into mechanisms for density segregation of granular mixtures in rotating cylinders, *Granular Matter* 13 (2011) 53–74.
- [32] G. Pereira, P. Cleary, Segregation due to particle shape of a granular mixture in a slowly rotating tumbler, *Granular Matter* 19 (2017) 23.
- [33] S. He, J. Gan, D. Pinson, A. Yu, Z. Zhou, Radial segregation of binary-sized ellipsoids in a rotating drum, *Powder Technology* 357 (2019) 322–330.
- [34] S. Kumar, S. Khatoon, J. Yogi, S. K. Verma, A. Anand, Experimental investigation of segregation in a rotating drum with non-spherical particles, *Powder Technology* 411 (2022) 117918.
- [35] S. Kumar, S. Khatoon, P. Dubey, J. Yogi, A. Anand, Shape-dependent radial segregation in rotating drum: Insights from DEM simulations, *Powder Technology* 432 (2024) 119134.
- [36] N. Thomas, Reverse and intermediate segregation of large beads in dry granular media, *Physical Review E* 62 (2000) 961.
- [37] D. Khakhar, A. V. Orpe, J. Ottino, Continuum model of mixing and size segregation in a rotating cylinder: Concentration-flow coupling and streak formation, *Powder Technology* 116 (2001) 232–245.

- [38] C. Liao, S. Hsiao, H. Nien, Density-driven spontaneous streak segregation patterns in a thin rotating drum, *Physical Review E* 89 (2014) 062204.
- [39] G. Pereira, S. Pucilowski, K. Liffman, P. Cleary, Streak patterns in binary granular media in a rotating drum, *Applied Mathematical Modelling* 35 (2011) 1638–1646.
- [40] G. Pereira, N. Tran, P. Cleary, Segregation of combined size and density varying binary granular mixtures in a slowly rotating tumbler, *Granular Matter* 16 (2014) 711–732.
- [41] W. Wu, K. Chen, E. Tsotsas, Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation, *Advanced Powder Technology* 33 (2022) 103512.
- [42] K. M. Hill, J. Kakalios, Reversible axial segregation of binary mixtures of granular materials, *Physical Review E* 49 (1994) R3610.
- [43] S. Fiedor, J. Ottino, Dynamics of axial segregation and coarsening of dry granular materials and slurries in circular and square tubes, *Physical Review Letters* 91 (2003) 244301.
- [44] G. Juarez, J. Ottino, R. Lueptow, Axial band scaling for bidisperse mixtures in granular tumblers, *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 78 (2008) 031306.
- [45] M. Arntz, W. K. den Otter, H. Beftink, R. Boom, W. J. Briels, The influence of end walls on the segregation pattern in a horizontal rotating drum, *Granular Matter* 15 (2013) 25–38.
- [46] K. M. Hill, J. Kakalios, Reversible axial segregation of rotating granular media, *Physical Review E* 52 (1995) 4393.
- [47] N. A. Pohlman, J. M. Ottino, R. M. Lueptow, End-wall effects in granular tumblers: From quasi-two-dimensional flow to three-dimensional flow, *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 74 (2006) 031305.
- [48] P. Chen, J. M. Ottino, R. M. Lueptow, Onset mechanism for granular axial band formation in rotating tumblers, *Physical Review Letters* 104 (2010) 188002.
- [49] Z. Cui, Y. Zhao, Y. Chen, X. Liu, Z. Hua, C. Zhou, J. Zheng, Transition of axial segregation patterns in a long rotating drum, *Particuology* 13 (2014) 128–133.
- [50] P. A. Cundall, O. D. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1979) 47–65.

- [51] E. Tsotsas, Particle-particle heat transfer in thermal DEM: Three competing models and a new equation, *International Journal of Heat and Mass Transfer* 132 (2019) 939–943.
- [52] J. Fischer, S. J. Rodrigues, M. Kriegeskorte, N. Hilse, E. Illana, V. Scherer, E. Tsotsas, Particle-particle contact heat transfer models in thermal DEM: A model comparison and experimental validation, *Powder Technology* 429 (2023) 118909.
- [53] A. Kharaghani, T. Metzger, E. Tsotsas, A proposal for discrete modeling of mechanical effects during drying, combining pore networks with DEM, *AIChE Journal* 57 (2011) 872–885.
- [54] M. Börner, A. Bück, E. Tsotsas, DEM-CFD investigation of particle residence time distribution in top-spray fluidised bed granulation, *Chemical Engineering Science* 161 (2017) 187–197.
- [55] Z. Jiang, C. Rieck, A. Bück, E. Tsotsas, Modeling of inter-and intra-particle coating uniformity in a Wurster fluidized bed by a coupled CFD-DEM-Monte Carlo approach, *Chemical Engineering Science* 211 (2020) 115289.
- [56] V. Brandt, J. Grabowski, N. Jurtz, M. Kraume, H. Kruggel-Emden, A benchmarking study of different DEM coarse graining strategies, *Powder Technology* 426 (2023) 118629.
- [57] V. Brandt, J. Grabowski, N. Jurtz, M. Kraume, H. Kruggel-Emden, DEM and DEM-CFD modeling of systems with geometric constrictions using a new particle location based multi-level coarse graining approach, *Powder Technology* 436 (2024) 119447.
- [58] G. Lu, J. Third, C. Müller, Critical assessment of two approaches for evaluating contacts between super-quadric shaped particles in DEM simulations, *Chemical Engineering Science* 78 (2012) 226–235.
- [59] N. Govender, D. N. Wilke, S. Kok, Collision detection of convex polyhedra on the NVIDIA GPU architecture for the discrete element method, *Applied Mathematics and Computation* 267 (2015) 810–829.
- [60] A. Podlozhnyuk, S. Pirker, C. Kloss, Efficient implementation of superquadric particles in discrete element method within an open-source framework, *Computational Particle Mechanics* 4 (2017) 101–118.
- [61] R. Berger, C. Kloss, A. Kohlmeyer, S. Pirker, Hybrid parallelization of the LIGGGHTS open-source DEM code, *Powder Technology* 278 (2015) 234–247.

- [62] J. Gan, Z. Zhou, A. Yu, A GPU-based DEM approach for modelling of particulate systems, *Powder Technology* 301 (2016) 1172–1182.
- [63] N. Govender, D. N. Wilke, S. Kok, Blaze-DEMGPU: Modular high performance DEM framework for the GPU architecture, *SoftwareX* 5 (2016) 62–66.
- [64] C. Kloss, C. Goniva, A. Hager, S. Amberger, S. Pirker, Models, algorithms and validation for opensource DEM and CFD-DEM, *Progress in Computational Fluid Dynamics* 12 (2012) 140–152.
- [65] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. In't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, et al., LAMMPS—a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Computer Physics Communications* 271 (2022) 108171.
- [66] S. B. Yeom, E.-S. Ha, M.-S. Kim, S. H. Jeong, S.-J. Hwang, D. H. Choi, Application of the discrete element method for manufacturing process simulation in the pharmaceutical industry, *Pharmaceutics* 11 (2019) 414.
- [67] W. R. Ketterhagen, M. T. am Ende, B. C. Hancock, Process modeling in the pharmaceutical industry using the discrete element method, *Journal of Pharmaceutical Sciences* 98 (2009) 442–470.
- [68] H. A. Navarro, M. P. de Souza Braun, Determination of the normal spring stiffness coefficient in the linear spring-dashpot contact model of discrete element method, *Powder Technology* 246 (2013) 707–722.
- [69] A. Di Renzo, F. P. Di Maio, Comparison of contact-force models for the simulation of collisions in DEM-based granular flow codes, *Chemical Engineering Science* 59 (2004) 525–541.
- [70] Y. Tsuji, T. Tanaka, T. Ishida, Lagrangian numerical simulation of plug flow of cohesionless particles in a horizontal pipe, *Powder Technology* 71 (1992) 239–250.
- [71] K. F. Malone, B. H. Xu, Determination of contact parameters for discrete element method simulations of granular systems, *Particuology* 6 (2008) 521–528.
- [72] D. Antypov, J. Elliott, On an analytical solution for the damped Hertzian spring, *Europhysics Letters* 94 (2011) 50004.
- [73] R. D. Mindlin, Compliance of elastic bodies in contact, *Journal of Applied Mechanics* 16 (1949) 259–268.

- [74] A. Di Renzo, F. P. Di Maio, An improved integral non-linear model for the contact of particles in distinct element simulations, *Chemical Engineering Science* 60 (2005) 1303–1312.
- [75] J. Ai, J.-F. Chen, J. M. Rotter, J. Y. Ooi, Assessment of rolling resistance models in discrete element simulations, *Powder Technology* 206 (2011) 269–282.
- [76] F. Fraige, P. Langston, Integration schemes and damping algorithms in distinct element models, *Advanced Powder Technology* 15 (2004) 227–245.
- [77] H. Kruggel-Emden, M. Sturm, S. Wirtz, V. Scherer, Selection of an appropriate time integration scheme for the discrete element method (DEM), *Computers & Chemical Engineering* 32 (2008) 2263–2279.
- [78] L. Verlet, Computer “experiments” on classical fluids, Part 1: Thermodynamical properties of Lennard-Jones molecules, *Physical Review* 159 (1967) 98.
- [79] W. C. Swope, H. C. Andersen, P. H. Berens, K. R. Wilson, A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters, *The Journal of Chemical Physics* 76 (1982) 637–649.
- [80] W. Zhong, A. Yu, X. Liu, Z. Tong, H. Zhang, DEM/CFD-DEM modelling of non-spherical particulate systems: theoretical developments and applications, *Powder Technology* 302 (2016) 108–152.
- [81] H. Kruggel-Emden, S. Rickelt, S. Wirtz, V. Scherer, A study on the validity of the multi-sphere discrete element method, *Powder Technology* 188 (2008) 153–165.
- [82] G. Lu, J. Third, C. Müller, Discrete element models for non-spherical particle systems: From theoretical developments to applications, *Chemical Engineering Science* 127 (2015) 425–465.
- [83] A. Džiugys, B. Peters, An approach to simulate the motion of spherical and non-spherical fuel particles in combustion chambers, *Granular Matter* 3 (2001) 231–266.
- [84] H. Ma, L. Zhou, Z. Liu, M. Chen, X. Xia, Y. Zhao, A review of recent development for the cfd-dem investigations of non-spherical particles, *Powder Technology* 412 (2022) 117972.
- [85] D. J. Evans, S. Murad, Singularity free algorithm for molecular dynamics simulation of rigid polyatomics, *Molecular Physics* 34 (1977) 327–331.

- [86] P. A. Cundall, Formulation of a three-dimensional distinct element model, Part 1: A scheme to detect and represent contacts in a system composed of many polyhedral blocks, in: *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, volume 25, Elsevier, 1988, pp. 107–116.
- [87] E. G. Nezami, Y. M. Hashash, D. Zhao, J. Ghaboussi, A fast contact detection algorithm for 3-D discrete element method, *Computers and Geotechnics* 31 (2004) 575–587.
- [88] C. W. Boon, G. Houlsby, S. Utili, A new algorithm for contact detection between convex polygonal and polyhedral particles in the discrete element method, *Computers and Geotechnics* 44 (2012) 73–82.
- [89] A. H. Barr, Superquadrics and angle-preserving transformations, *IEEE Computer Graphics and Applications* 1 (1981) 11–23.
- [90] S. Wang, D. Marmysh, S. Ji, Construction of irregular particles with superquadric equation in DEM, *Theoretical and Applied Mechanics Letters* 10 (2020) 68–73.
- [91] A. Jaklic, A. Leonardis, F. Solina, *Segmentation and Recovery of Superquadrics*, Springer Science & Business Media, 2000.
- [92] A. Jaklic, F. Solina, Moments of superellipsoids and their application to range image registration, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 33 (2003) 648–657.
- [93] O. R. Walton, R. L. Braun, Simulation of rotary-drum and repose tests for frictional spheres and rigid sphere clusters, Technical Report, Lawrence Livermore National Lab., CA (United States), 1993.
- [94] I. P. Omelyan, Numerical integration of the equations of motion for rigid polyatomics: The matrix method, *Computer Physics Communications* 109 (1998) 171–183.
- [95] I. P. Omelyan, A new leapfrog integrator of rotational motion. the revised angular-momentum approach, *Molecular Simulation* 22 (1999) 213–236.
- [96] T. Miller Iii, M. Eleftheriou, P. Pattnaik, A. Ndirango, D. Newns, G. Martyna, Symplectic quaternion scheme for biophysical molecular dynamics, *The Journal of Chemical Physics* 116 (2002) 8649–8659.
- [97] L. Wang, J.-Y. Park, Y. Fu, Representation of real particles for DEM simulation using X-ray tomography, *Construction and Building Materials* 21 (2007) 338–346.
- [98] J.-F. Ferrellec, G. R. McDowell, A method to model realistic particle shape and inertia in DEM, *Granular Matter* 12 (2010) 459–467.

- [99] H. P. Zhu, Z. Y. Zhou, R. Yang, A. Yu, Discrete particle simulation of particulate systems: theoretical developments, *Chemical Engineering Science* 62 (2007) 3378–3396.
- [100] S. J. Burns, P. T. Piiroinen, K. J. Hanley, Critical time step for DEM simulations of dynamic systems using a Hertzian contact model, *International Journal for Numerical Methods in Engineering* 119 (2019) 432–451.
- [101] Y. Li, Y. Xu, C. Thornton, A comparison of discrete element simulations and experiments for “sandpiles” composed of spherical particles, *Powder Technology* 160 (2005) 219–228.
- [102] S. Plimpton, Fast parallel algorithms for short-range molecular dynamics, *Journal of Computational Physics* 117 (1995) 1–19.
- [103] Z. Yao, J.-S. Wang, G.-R. Liu, M. Cheng, Improved neighbor list algorithm in molecular simulations using cell decomposition and data sorting method, *Computer Physics Communications* 161 (2004) 27–35.
- [104] P. Auerkari, Mechanical and physical properties of engineering alumina ceramics, volume 23, Technical Research Centre of Finland Espoo, Finland, 1996.
- [105] H. Kruggel-Emden, E. Simsek, S. Rickelt, S. Wirtz, V. Scherer, Review and extension of normal force models for the discrete element method, *Powder Technology* 171 (2007) 157–173.
- [106] M. Yamamoto, S. Ishihara, J. Kano, Evaluation of particle density effect for mixing behavior in a rotating drum mixer by DEM simulation, *Advanced Powder Technology* 27 (2016) 864–870.
- [107] H. Nakashima, Y. Shioji, T. Kobayashi, S. Aoki, H. Shimizu, J. Miyasaka, K. Ohdoi, Determining the angle of repose of sand under low-gravity conditions using discrete element method, *Journal of Terramechanics* 48 (2011) 17–26.
- [108] H. Chen, Y. Xiao, Y. Liu, Y. Shi, Effect of Young’s modulus on DEM results regarding transverse mixing of particles within a rotating drum, *Powder Technology* 318 (2017) 507–517.
- [109] E. Alizadeh, F. Bertrand, J. Chaouki, Comparison of DEM results and lagrangian experimental data for the flow and mixing of granules in a rotating drum, *AIChE Journal* 60 (2014) 60–75.
- [110] R. G. Sherritt, J. Chaouki, A. K. Mehrotra, L. A. Behie, Axial dispersion in the three-dimensional mixing of particles in a rotating drum reactor, *Chemical Engineering Science* 58 (2003) 401–415.

- [111] E. Alizadeh, O. Dubé, F. Bertrand, J. Chaouki, Characterization of mixing and size segregation in a rotating drum by a particle tracking method, *AIChE Journal* 59 (2013) 1894–1905.
- [112] D. Parker, A. Dijkstra, T. Martin, J. Seville, Positron emission particle tracking studies of spherical particle motion in rotating drums, *Chemical Engineering Science* 52 (1997) 2011–2022.
- [113] L. Zhang, F. Weigler, V. Idakiev, Z. Jiang, L. Mörl, J. Mellmann, E. Tsotsas, Experimental study of the particle motion in flighted rotating drums by means of magnetic particle tracking, *Powder Technology* 339 (2018) 817–826.
- [114] M. Nakagawa, S. A. Altobelli, A. Caprihan, E. Fukushima, E. K. Jeong, Non-invasive measurements of granular flows by magnetic resonance imaging, *Experiments in Fluids* 16 (1993) 54–60.
- [115] K. Yamane, M. Nakagawa, S. Altobelli, T. Tanaka, Y. Tsuji, Steady particulate flows in a horizontal rotating cylinder, *Physics of Fluids* 10 (1998) 1419–1427.
- [116] M. Schutyser, J. Padding, F. Weber, W. Briels, A. Rinzema, R. Boom, Discrete particle simulations predicting mixing behavior of solid substrate particles in a rotating drum fermenter, *Biotechnology and Bioengineering* 75 (2001) 666–675.
- [117] M. Cho, P. Dutta, J. Shim, A non-sampling mixing index for multicomponent mixtures, *Powder Technology* 319 (2017) 434–444.
- [118] N. G. Deen, G. Willem, G. Sander, J. Kuipers, Numerical analysis of solids mixing in pressurized fluidized beds, *Industrial & Engineering Chemistry Research* 49 (2010) 5246–5253.
- [119] C. Xie, H. Ma, Y. Zhao, Investigation of modeling non-spherical particles by using spherical discrete element model with rolling friction, *Engineering Analysis with Boundary Elements* 105 (2019) 207–220.
- [120] M. K. Saeed, M. S. Siraj, Mixing study of non-spherical particles using DEM, *Powder Technology* 344 (2019) 617–627.

Academic Contributions

Journal publications

- 1) **Wencong Wu**, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of particle mixing time in a rotary drum by 2D DEM simulations and cross-correlation.” *Advanced Powder Technology* 33.4 (2022): 103512.
- 2) **Wencong Wu**, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of particle mixing in rotary drums by a DEM data-driven PSO-SVR model.” *Powder Technology* 434 (2024): 119365.
- 3) **Wencong Wu**, Kaicheng Chen, and Evangelos Tsotsas. “Prediction of rod-like particle mixing in rotary drums by three machine learning methods based on DEM simulation data.” *Powder Technology* 448 (2024): 120307.

Conference presentations

- 1) **Wencong Wu**, Kaicheng Chen, Michael Jacob, and Evangelos Tsotsas. Cross-correlation of mixedness between 3D and 2D DEM simulations for particle mixing in a rotary drum, In: *Jahrestreffen der ProcessNet-Fachgruppen Lebensmittelverfahrenstechnik, Mischvorgänge, Grenzflächenbestimmte Systeme und Prozesse*, Online, March, 2021

Conference posters

- 1) **Wencong Wu**, Kaicheng Chen, Michael Jacob, and Evangelos Tsotsas. Cross-correlation between 2D and 3D DEM simulation of particle mixing in the rotating drum, In: *Jahrestreffen der ProcessNet-Fachgruppen Lebensmittelverfahrenstechnik, Mischvorgänge, Grenzflächenbestimmte Systeme und Prozesse*, Würzburg (DE), March, 2020
- 2) **Wencong Wu**, Kaicheng Chen, and Evangelos Tsotsas. Prediction of mixedness from DEM simulations for particle mixing in a rotary drum by machine learning

methods, In: *Jahrestreffen der ProcessNet-Fachgruppen Computational Fluid Dynamics, Mischvorgänge und Agglomerations- und Schüttguttechnik*, Leipzig (DE), March, 2022

Supervised student theses

- 1) Krishna Ramya Pavuluri, Investigation on the agglomeration process of milk powder in fluidized bed system, Master thesis, Otto-von-Guericke-University Magdeburg, Germany, 2020.
- 2) Rui Wang, Investigation on particle mixing and segregation process in a rotating drum by different methods, Master thesis, Otto-von-Guericke-University Magdeburg, Germany, 2021.

Declaration of Honor

I hereby declare that I produced this thesis without prohibited external assistance and that none other than the listed references and tools have been used.

In the case of co-authorship, especially in the context of a cumulative dissertation, the own contribution is correctly and completely stated. I did not make use of any commercial consultant concerning graduation. A third party did not receive any nonmonetary perquisites neither directly nor indirectly for activities which are connected with the contents of the presented thesis. All sources of information are clearly marked, including my own publications.

In particular I have not consciously:

- Fabricated data or rejected undesired results
- Misused statistical methods with the aim of drawing other conclusions than those warranted by the available data
- Plagiarized data or publications
- Presented the results of other researchers in a distorted way

I do know that violations of copyright may lead to injunction and damage claims of the author and also to prosecution by the law enforcement authorities.

I hereby agree that the thesis may need to be reviewed with an electronic data processing for plagiarism.

This work has not yet been submitted as a doctoral thesis in the same or a similar form in Germany or in any other country. It has not yet been published as a whole.

Magdeburg, March 31, 2025

M.Sc. Wencong Wu