Engineering Applications of Artificial Intelligence 167 (2026) 113913

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Neural network-driven shape representation and computational particle mechanics via signed distance fields

Chenghao Li [a], Zhengshou Lai [a,b,c,*], Shuai Huang [b], Linchong Huang [a,b,c]

[a] School of Civil Engineering, Sun Yat-Sen University, Zhuhai 519082, China
[b] State Key Laboratory of Tunnel Engineering, Zhuhai 519082, China
[c] Guangdong Key Laboratory of Marine Civil Engineering, Sun Yat-sen University, Zhuhai, 519082, China

ARTICLE INFO

Keywords:
Irregular shape
Signed distance field
Neural network
Discrete element method
Granular material

ABSTRACT

This study presents a framework for shape representation and computational particle mechanics of granular materials using neural network-encoded signed distance fields. The approach leverages a neural network to learn and represent a signed distance field, mapping spatial points to their signed distance from the particle surface. Two neural network models are explored: one incorporating a latent code to capture shape variations, and the other without such encoding. The accuracy of these models in capturing particle morphology is rigorously evaluated, and their ability to generate new particles with realistic shapes is demonstrated. The proposed neural network-based approach is seamlessly integrated into the signed distance field-based discrete element method, enabling efficient and robust modeling of granular particles with arbitrary shapes. The integration is validated through discrete element-based simulations, demonstrating its effectiveness in particle mechanics applications. Additionally, memory consumption and computational performance are analyzed. These contributions position the neural network-encoded signed distance fields framework as a versatile and powerful tool for advancing computational modeling of granular materials.

1. Introduction

The mechanical behavior of granular materials is strongly governed by the shape of individual particles, which influences interparticle contact, packing configuration, and macroscopic response (Cho et al., 2006; Carrasco et al., 2025; Chen et al., 2025; Cúñez et al., 2024). Accurate and efficient representation of particle morphology is therefore essential for predictive modeling of granular materials, particularly in contexts involving anisotropic, angular, or irregular particles. Among various numerical methods, the discrete element method (DEM) (Cundall and Strack, 1979) has become the predominant approach for simulating particle-scale interactions in granular assemblies due to its ability to explicitly resolve contact dynamics. As such, DEM-based simulations increasingly rely on high-fidelity geometric representations to capture shape-dependent phenomena such as shear dilation, interlocking, and flow segregation. This has motivated extensive efforts in both experimental shape acquisition (Blott and Pye, 2008; Zhou et al., 2015; Yin et al., 2023; Xiong et al., 2024) and computational modeling frameworks (Zhao et al., 2023b; Feng, 2023). However, effectively incorporating complex geometries into large-scale DEM remains an open research direction, particularly in applications involving highly irregular shapes, evolving morphologies, or coupled multiphysics interactions.

Within the DEM, particle shape modeling refers to the mathematical formulation used to describe particle geometry for contact detection, force evaluation, and kinematic update. Existing modeling strategies can be broadly classified into surface-based and volume-based representations, depending on how particle geometry is defined. Surface-based approaches describe particle boundaries explicitly. Typical examples include analytical functions such as spheres, ellipsoids, and spherical harmonics (Wang et al., 2021; Zhao and Zhao, 2019; Lai and Huang, 2021; Radvilaitė et al., 2016), as well as combinational representations constructed from simpler primitives, including clumps and metaballs (Zhao et al., 2023b; Zhang et al., 2021). Another widely used surface-based option is discretized boundary descriptions, most notably triangle meshes (Zhan et al., 2021), which capture angularity and sharp geometric features and are often combined with advanced contact algorithms such as Minkowski-sum-based formulations (Wang

* Correspondence to: Sun Yat-sen University, Tangjiawan, Xiangzhou District, Zhuhai, 519082, Guangdong, China.
E-mail addresses: lichh53@mail2.sysu.edu.cn (C. Li), laizhengsh@mail.sysu.edu.cn (Z. Lai), huangsh283@mail.sysu.edu.cn (S. Huang), hlinch@mail.sysu.edu.cn (L. Huang).

https://doi.org/10.1016/j.engappai.2026.113913
Received 6 August 2025; Received in revised form 7 January 2026; Accepted 18 January 2026

et al., 2024). Volume-based approaches, in contrast, define particle geometry through explicit volumetric occupancy. Representative examples include polyhedral models (Qiao et al., 2025) and voxel-based representations (van der Haven et al., 2023), which facilitate mass and inertia evaluation and coupling with continuum fields, but typically incur higher memory and computational costs. While explicit surface and volume models have enabled high-fidelity representation of irregular particles, their scalability and robustness in large-scale DEM simulations remain constrained by contact complexity, storage requirements, and geometric consistency across different shape families.

Beyond representing individual particles, many DEM applications require the synthesis of large particle populations with controlled morphological variability, motivating the development of virtual generation techniques that are distinct from direct shape reconstruction. Existing virtual generation methods can be broadly divided into two categories: topological evolution and parametric mapping. Topological evolution approaches generate new particles by stochastic modification or aggregation of geometric primitives, allowing shape diversity to emerge through structural variation. Examples include Fourier-descriptor-based perturbation of particle outlines (Mollon and Zhao, 2013), clone-based evolution schemes derived from reference particle libraries (Jerves et al., 2017), and metaball-based overlapping constructions that approximate complex morphologies through topological summation (Zhao et al., 2023a). Parametric mapping methods instead rely on dimensionality reduction of shape descriptors, establishing a mapping between high-dimensional modeling parameters and a low-dimensional latent space. New particle shapes are generated by sampling in this latent space and projecting back to the modeling domain. A representative example is the morphological gene framework (Xiong and Wang, 2021; Xiong et al., 2022), which operates on spherical harmonics coefficients to describe shape mutation and decay while retaining a fixed underlying modeling representation. In recent years, data-driven and machine-learning techniques have been increasingly incorporated into both reconstruction and generation pipelines, enabling latent-space learning, statistical shape synthesis, and image-based reconstruction (Zhu et al., 2025; Shi et al., 2021; Argilaga et al., 2024; Xi et al., 2025; Zhang and Willis, 2024). However, most existing virtual generation strategies remain tightly coupled to specific explicit shape models or discretized geometries, limiting their direct compatibility with contact-intensive DEM simulations. This gap highlights the need for a representation framework that can simultaneously support flexible generation, compact encoding, and physically consistent geometric queries.

As an alternative to traditional shape representations, signed distance field (SDF) has emerged as a unified and versatile implicit modeling paradigm. By defining geometry as continuous scalar fields, SDF enable efficient distance and gradient evaluations, which are well suited for contact-resolving simulations such as DEM. Early implementations have demonstrated the advantages of level set-based SDF for complex contact modeling and shape evolution in granular systems (Kawamoto et al., 2016, 2018; Duriez and Bonelli, 2021; Moncada et al., 2023; Tan and Sitar, 2024). Recent studies have further developed SDF-based DEM frameworks that integrate shape-contact interactions via generic interfaces, support energy-conserving formulations, and generalize multiple analytical shape models within a unified formulation (Lai et al., 2022b, 2024). These frameworks have been successfully applied to simulate irregular particle assemblies (Huang et al., 2023) and fluid-solid interaction scenarios (Lai et al., 2023), with extensions exploring machine learning-enabled contact detection (Huang et al., 2024). Together, these efforts highlight the flexibility, robustness, and physical consistency offered by SDF-based modeling. Nevertheless, most current SDF implementations rely on analytically constructed or procedurally defined particle shapes, limiting their adaptability in large-scale or data-driven simulations. For applications involving diverse, evolving, or experimentally derived morphologies, manually specifying shape descriptors becomes a major bottleneck. These limitations point to the need for scalable, learnable representations that can compactly encode and reuse complex geometries.

In recent years, neural implicit models have gained significant attention in the graphics and vision communities for representing three dimensional (3D) geometry using continuous fields, such as signed distance functions or occupancy values, learned through neural networks. These models offer compactness, smooth interpolation, and topological flexibility. Representative techniques include DeepSDF (Park et al., 2019), MetaSDF (Sitzmann et al., 2020), and occupancy networks (Mescheder et al., 2019), which learn mappings from spatial coordinates to shape indicators (Chen and Zhang, 2019; Chibane et al., 2020). Further developments introduced multiresolution decoders (Chibane and Pons-Moll, 2020), differentiable rendering (Niemeyer et al., 2020), signed-agnostic learning (Atzmon and Lipman, 2020a), semantic priors (Zheng et al., 2021), and geometric regularization methods (Gropp et al., 2020; Atzmon and Lipman, 2020b). Recent efforts have extended these models using convolutional backbones (Peng et al., 2020), diffusion-based generative frameworks (Li et al., 2023), radiance-field based representations such as NeRF (Mildenhall et al., 2021) and its variants (Barron et al., 2021), as well as 3D Gaussian splatting (Kerbl et al., 2023; Wu et al., 2024). In particular, NeRF represents a scene as a continuous radiance and density field and renders novel views via differentiable volumetric integration along camera rays, whereas 3D Gaussian splatting models a scene using a set of anisotropic Gaussian primitives and employs rasterization-based splatting for efficient, real-time rendering. Despite their success in shape reconstruction and generation, these methods have yet to be fully integrated into physics-based modeling workflows. In particular, their application in contact-resolving solvers such as DEM remains limited, and simulation-oriented requirements, such as fast distance queries, shape-conditional evaluation, and integration with mechanical formulations, are not inherently addressed by existing neural implicit pipelines. Moreover, their potential for supporting inverse design, virtual particle generation, and shape topology optimization in granular systems remains largely unexplored.

To address these challenges, we develop and explore NetSDF, a unified framework for learning and utilizing neural network-encoded signed distance fields to represent granular particle shapes. By combining the geometric rigor of SDFs with the adaptability of neural implicit models, the proposed approach enables compact, differentiable, and generative shape representations suitable for both simulation and design-oriented tasks. The main contributions of this work are as follows: (1) development of neural SDF models with and without latent encoding for flexible and reusable particle shape representation; (2) integration of NetSDF into an SDF-DEM framework, enabling accurate and robust simulation of granular assemblies with arbitrarily shaped particles; and (3) comprehensive evaluation of computational performance and shape modeling capabilities, demonstrating the framework's effectiveness in both simulation and virtual shape generation tasks.

The remainder of this paper is organized as follows. Section 2 presents the methodologies of SDF-DEM, the NetSDF framework (SingleNetSDF), and its latent encoding variant (CodedNetSDF) for shape representation and computational mechanics. Section 3 validates the proposed framework through numerical examples, specifically focusing on its effectiveness in particle shape reconstruction and generation. Section 4 investigates the influence of network architecture and hyperparameters on model performance, and evaluates memory usage and computational efficiency. Subsequently, Section 5 presents the practical application of the framework in DEM simulations, demonstrating its capability in modeling granular assemblies with complex morphologies. Finally, Section 6 summarizes the findings and outlines directions for future research.

2. Methodologies

This section introduces the neural network-based particle models, NetSDF, for particle shape representation and computational particle

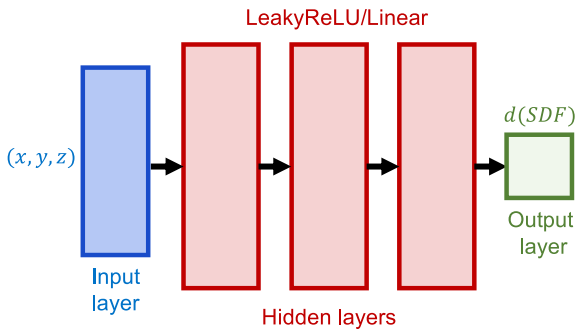


Fig. 1. Neural network-based SDF model for a single particle (SingleNetSDF).

mechanics in granular materials. It outlines the neural network architecture for encoding SDFs of particles, reconstructing particle surfaces from these encoded fields, and generating virtual particles through neural networks. Additionally, the integration of the proposed NetSDF particle models into the SDF-DEM framework is discussed.

2.1. Brief introduction to SDF-DEM

The SDF-DEM framework (Lai et al., 2022b) represents particles using an SDF, a continuous function that maps any spatial point to a signed value. Unlike the conventional SDF definition, which typically represents the shortest distance from a query point to the particle surface, the SDF-DEM framework adopts a more generalized approach. This approach allows for the use of alternative distance metrics, such as the shortest or radial distance, as long as the resulting SDF isosurfaces remain non-self-intersecting. In this framework, the signed distance is positive for points inside the particle and negative for points outside, with the zero isosurface (where the signed distance equals zero) precisely defining the particle surface. In addition to the SDF function, the SDF-based particle model incorporates a surface projection function (SPF), which projects points within the particle onto its surface. This function plays a critical role in contact detection, enabling the identification of contact points required to compute contact forces and moments in DEM simulations.

Contact detection in SDF-DEM employs a node-to-surface algorithm, where the surface of one particle is discretized into a set of surface nodes. The algorithm determines whether any of these nodes intrude into another particle. Once contact is detected, the SDF-DEM framework calculates the normal contact force using an energy-conserving contact theory (Feng, 2021). The tangential contact force is resolved with the classical linear spring model incorporating Coulomb friction. This combination ensures both accuracy and computational efficiency in detecting and resolving particle contacts during granular material simulations. For a detailed discussion of the SDF-DEM framework, including technical specifications and implementation, readers are referred to Lai et al. (2022b, 2024).

2.2. Neural network-based SDF

In the SDF-DEM framework, creating a new particle shape model requires the definition of both the SDF and the SPF. A series of particle models based on conventional shape descriptions (e.g., sphere, ellipsoid, spherical harmonics, level set, polyhedron) has been previously developed in Lai et al. (2022b). In this study, a data-driven approach is introduced, where machine learning techniques are employed to learn implicit representations of SDFs directly, enabling flexible and accurate modeling of complex particle geometries.

As illustrated in Fig. 1, the neural network model takes a query position as input and predicts the corresponding SDF value as output. The input layer consists of three neurons representing the spatial coordinates of a query point. This is followed by several hidden layers with non-linear activation functions (e.g., LeakyReLU (Xu et al., 2020)) to model the complex spatial relationships inherent in SDF representations. The output layer contains a single neuron with a linear activation function to predict the SDF value. This design balances computational efficiency with the capacity to capture the intricate spatial dependencies required for accurate SDF representation.

With the neural network-based SDF, two approaches, namely the gradient-decent and ray-searching methods, are proposed to obtain the corresponding surface point of a query point (i.e., the SPF for the SDF-DEM framework (Lai et al., 2022b)).

- *Gradient-descent approach*: The gradient-descent approach leverages the spatial gradient of the SDF, which is obtained via automatic differentiation of the neural SDF network, to iteratively refine the surface projection point (see Algorithm 1). The initial position of the projection point is set to the intruding query point $\mathbf{q}$. During each iteration, the projection point $\mathbf{p}$ is updated along the negative gradient direction of the SDF, driving it toward the particle surface. This method offers fast convergence by exploiting the local geometric information encoded in the neural SDF. However, when the network-defined SDF exhibits tortuous or locally non-smooth characteristics, convergence issues or slower progress toward the true surface may arise. To mitigate these challenges, stabilization strategies such as adaptive step sizes can be employed.

Algorithm 1 Gradient-descent surface projection

Require: Query point $\mathbf{q}$, SDF $\phi(\mathbf{x})$, gradient $\nabla\phi(\mathbf{x})$, tolerance ϵ, maximum iterations N
Ensure: Projection point $\mathbf{p}$
$\mathbf{p} \leftarrow \mathbf{q}$ ▷ Initialize at query point
for $k = 1$ to N **do**
  $\mathbf{g} \leftarrow \nabla\phi(\mathbf{p})$ ▷ Compute SDF gradient via auto-diff
  if $\|\mathbf{g}\| < \epsilon$ **or** $|\phi(\mathbf{p})| < \epsilon$ **then**
    break ▷ Converged if gradient small or near surface
  end if
  $\alpha \leftarrow$ adaptive_step_size($\mathbf{p}, \mathbf{g}$) ▷ Adaptive step for stability
  $\mathbf{p} \leftarrow \mathbf{p} - \alpha\mathbf{g}$ ▷ Move along negative gradient
end for
return $\mathbf{p}$

Note: The SDF $\phi(\mathbf{p})$ is the direct output of the neural network, while its spatial gradient $\nabla\phi(\mathbf{p})$ is computed via automatic differentiation of the network with respect to the input coordinates.

- *Ray-searching approach*: The ray-searching approach refines the surface projection point by adjusting its position along a radial vector from the origin directing to the query point (see Algorithm 2). The displacement magnitude is directly proportional to the predicted SDF value, effectively moving the point closer to or farther from the particle surface. This process is repeated iteratively, with each adjustment reducing the discrepancy between the candidate point and the target surface, until the candidate point aligns with the desired surface. Compared to the gradient-descent approach, the ray-searching method is computationally simpler as it does not require the calculation of partial derivatives. This simplicity contributes to its robustness, particularly when dealing with complex or irregular surface geometries where the gradient-descent method might struggle with convergence issues. However, the ray-searching approach may require additional iterations to achieve the same level of precision as the gradient-descent method for fine surface details.

Algorithm 2 Ray-searching surface projection

Require: Query point $\mathbf{q}$, SDF $\phi(\mathbf{x})$, tolerance ϵ, scaling factor λ, maximum iterations N
Ensure: Projection point $\mathbf{p}$

$\hat{\mathbf{r}} \leftarrow \frac{\mathbf{q}}{\|\mathbf{q}\|}$ ▷ Fixed ray from origin to query point
$\mathbf{p} \leftarrow \mathbf{q}$ ▷ Initialize at query point
for $k = 1$ to N **do**
  $d \leftarrow \phi(\mathbf{p})$ ▷ Evaluate SDF at current point
  if $|d| < \epsilon$ **then**
    break ▷ Projection converged to surface
  end if
  $\mathbf{p} \leftarrow \mathbf{p} - \lambda d\hat{\mathbf{r}}$ ▷ Move along ray direction by SDF distance
end for
return p

2.3. Neural network-based SDF with latent code

To extend the SingleNetSDF model beyond fixed particle geometries and enable generative capabilities, we develop a conditional neural network architecture termed CodedNetSDF. While SingleNetSDF provides a compact and efficient representation of individual particle shapes, it lacks the capacity to generalize across morphological variations. In contrast, CodedNetSDF introduces a latent code as an additional input to the network, serving as a compact and expressive descriptor of shape-specific features. This conditioning mechanism enables the network to capture statistical variability within a particle dataset and to generate new, synthetic geometries while preserving the mathematical rigor of the signed distance field representation. The two variants of the proposed framework will hereafter be referred to as SingleNetSDF (without latent code) and CodedNetSDF (with latent code), respectively.

In the CodedNetSDF model, each template is assigned a unique latent code that encapsulates its geometric features. These latent codes are concatenated with spatial coordinates to form the input features for the neural network. Similar to the SingleNetSDF, the CodedNetSDF network adopts a fully connected feed-forward architecture, as illustrated in Fig. 2. The input layer has a dimension of $n_z + 3$, where n_z represents the dimension of the latent code. Hidden layers employ nonlinear activation functions to capture intricate dependencies, and the output layer predicts the SDF value. The training process involves the simultaneous optimization of the neural network parameters and latent codes. While the network learns the mapping from input features to SDF values, the latent codes are iteratively refined to better capture template-specific geometries. This dual optimization process enables the framework to effectively accommodate shape variations across multiple particle templates, ensuring both flexibility and accuracy in particle representation. The numerical implementation and training process will be detailed in a subsequent section.

2.4. Shape reconstruction from NetSDF

With the particle shape implicitly represented by NetSDF, surface mesh reconstruction may be necessary for visualization. The reconstruction process begins with the initialization of surface points, which can be generated using the SPF applied to random samples of query points. To ensure a uniform distribution of surface points, the spherical Voronoi sampling algorithm (Lai et al., 2022b) could be employed to generate the initial random samples. Subsequently, two reconstruction techniques, alpha shape approach and spherical harmonics approach, can be utilized to produce the final surface mesh. Fig. 3 illustrates the technical workflows of these two reconstruction approaches.

- *Alpha shape reconstruction*: Alpha shape reconstruction is a computational geometry method for reconstructing the shape of a point cloud or a discrete set of points in space. It extends the convex hull to capture both convex and concave shapes, with the level of detail controlled by the parameter α. In this work, the open-source library *CGAL* is used to implement alpha shape reconstruction (Project, 2024). An α value of 0.7 is selected, as it effectively captures intricate and concave geometries. The output is a triangulated mesh closely aligned with the refined surface points, making this technique well-suited for geometries with sharp features or irregular boundaries.
- *Spherical harmonics reconstruction*: Spherical harmonics provide an alternative approach, producing a smooth and continuous representation of the shape (Zhou et al., 2015). This method fits the surface points using a series of spherical harmonic base functions, offering a robust mathematical framework for generating smooth and symmetric representations. It is particularly advantageous for objects that are smooth or exhibit radial symmetry, resulting in a visually appealing, high-quality surface reconstruction. In this work, spherical harmonics of order 10 are used, striking a balance between computational efficiency and reconstruction accuracy. Once the spherical harmonic coefficients are determined, the surface mesh is obtained by first sampling points on a sphere, defined by polar and azimuthal angles, and constructing the convex hull of these points. The parametric points in the convex hull are then replaced with the corresponding surface points to produce the final surface mesh.

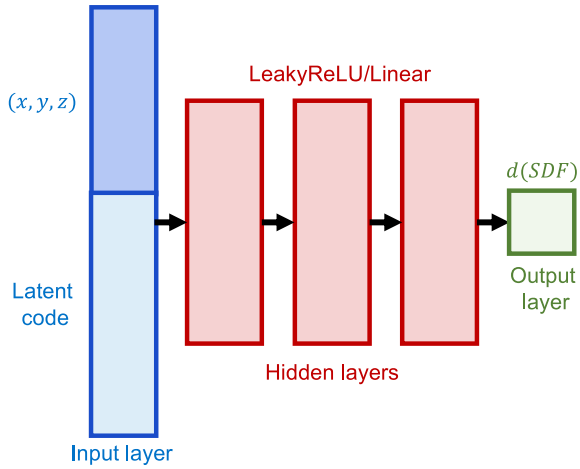


Fig. 2. Neural network-based SDF model with latent codes for multiple particles (CodedNetSDF).

Through these steps, a complete technical pipeline is established, enabling the mutual conversion between particle surface meshes, signed distance fields, and surface point clouds, as illustrated in Fig. 4.

2.5. Virtual shape generation from NetSDF

The NetSDF framework enables the generation of virtual particle geometries by exploiting the structured latent space learned during training. As illustrated in Fig. 5, geometric variations are encoded in a compact and continuous manner, allowing new particle shapes to be synthesized through latent space sampling. One practical strategy involves analyzing the latent codes within a given particle category to estimate their statistical distribution, from which new samples can be drawn. These synthesized particles preserve category-level morphological traits while introducing controllable variations beyond the original dataset. Such capability is particularly suited for constructing large virtual shape libraries, conducting uncertainty quantification, and augmenting data for simulation-driven studies. The resulting particles offer both structural plausibility and diversity, making them valuable for analyzing shape-dependent effects in granular flow, packing, or contact-driven mechanics.

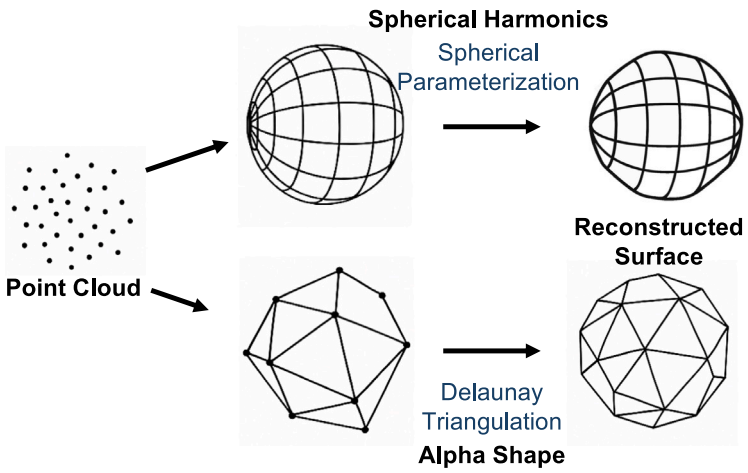


Fig. 3. Illustration of alpha shape and spherical harmonics approaches for surface reconstruction.

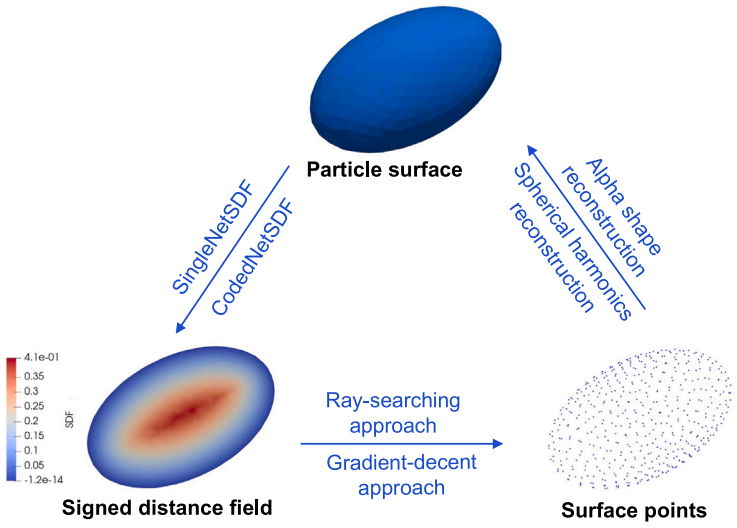


Fig. 4. Pipeline for converting between surface meshes, point clouds, and signed distance fields.

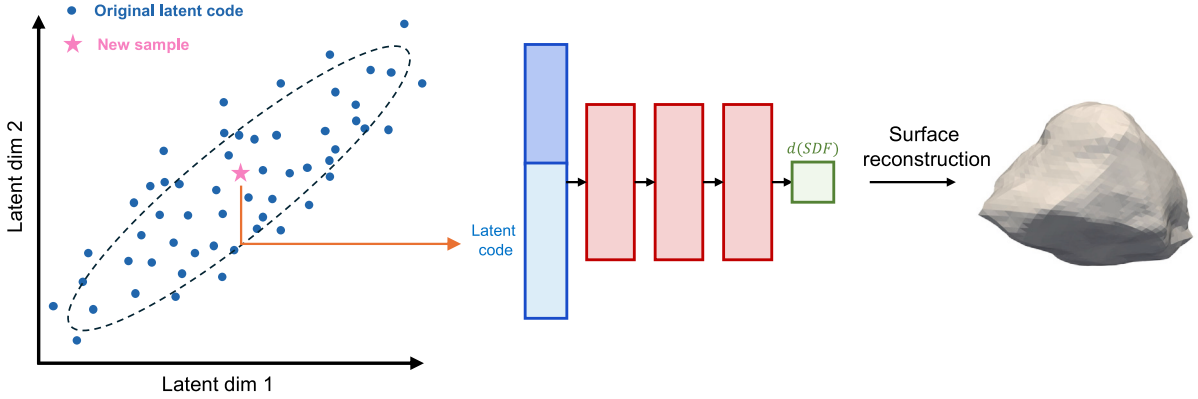


Fig. 5. Illustration of virtual particle shape generation via the random sampling approach.

Beyond sampling, NetSDF also supports shape interpolation by linearly blending between two latent codes, enabling smooth transitions between reference geometries, as shown in Fig. 6. This approach allows for continuous exploration along morphing paths in the latent space, producing intermediate shapes with blended features. It is particularly useful for performance-oriented shape optimization: given two particles with favorable properties, interpolation provides a systematic way to generate candidate shapes that combine desirable traits. These interpolated instances can then be evaluated through physics-based simulations to assess functional performance. This process enables iterative

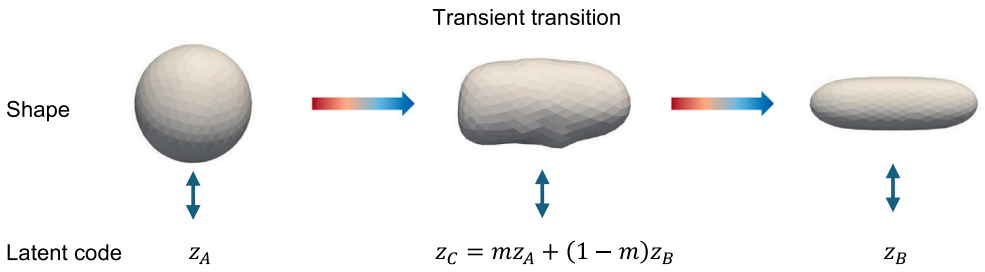


Fig. 6. Illustration of virtual particle shape generation via the interpolation approach.

refinement of particle geometry using a compact, low-dimensional, and expressive representation. Together, random sampling and interpolation form a generative modeling framework that integrates seamlessly with numerical simulations, offering a powerful tool for virtual testing, parametric exploration, and inverse design of granular materials.

2.6. Numerical implementation

The proposed NetSDF particle models are implemented using the open-source library *mlpack* (Curtin et al., 2023), a high-performance, flexible, and easy-to-use C++ machine learning library that supports a wide variety of algorithms. One of the key features of *mlpack* is its ability to efficiently handle large datasets and scale across multi-core systems, which is crucial for training complex models like NetSDF that require high computational efficiency. The use of *mlpack* provides a robust and efficient way to train and optimize neural network models, leveraging its automatic differentiation capabilities for gradient-based optimization. For the SingleNetSDF particle model, *mlpack* is used to implement a fully connected feed-forward network, tailored to predict the SDF values based on spatial coordinates. For the CodedNetSDF particle model, each particle template is associated with a latent code that encapsulates template-specific geometric information.

The training of SingleNetSDF follows a conventional back-propagation algorithm. In contrast, the training of CodedNetSDF is more complex as it requires optimizing both the model parameters and the latent code. The process begins by initializing the latent code from a uniform distribution, $\mathcal{U}(-1,1)$. Then, the model parameters and latent code are optimized through an iterative update process. In each iteration, the model parameters are updated using back-propagation. Following this, the latent codes are adjusted based on their contribution to the overall loss. The update rule for the latent code is as follows:

$$z_i \leftarrow z_i - \eta \cdot \nabla_{z_i} \mathcal{L} \tag{1}$$

where η is the learning rate for the latent codes, and $\nabla_{z_i}\mathcal{L}$ represents the gradient of the loss with respect to the latent codes. To maintain stability during training, the latent codes are updated with a learning rate of $\eta = 0.001$ and are clamped within the range $[-10, 10]$ in this work. This iterative optimization ensures that both the neural network and the latent codes effectively capture the geometric features of the particle templates, leading to accurate SDF representations.

The next step involves integrating the NetSDF models into *netdem*, an in-house simulation code for granular materials that combines DEM modeling with a machine learning environment (Lai et al., 2022a). The *netdem* is structured around an object-oriented programming approach, with core classes such as *Scene*, *Particle*, *Shape*, *Solver*, and *Modifier*. The *Shape* class handles the particle shapes in DEM simulations. It acts as a parent class, defining common properties like *id*, *volume*, *surface area*, and *axis-aligned bounding boxes* for all particle shapes, as illustrated in Fig. 7. Specialized subclasses for different shape representations, such as *Trimesh* (for triangle meshes), *LevelSet* (for voxel-based representations), and *SingleNetSDF/CodedNetSDF* (for neural network-based shapes), extend this parent class. This flexible design allows *netdem* to handle a variety of particle shapes within the same simulation environment. The SingleNetSDF and CodedNetSDF models are integrated as subclasses of the *Shape* class within the *netdem* framework. For the *SingleNetSDF* representation, each shape instance stores a single trained neural network that directly defines the SDF, without the use of a latent code. In contrast, the *CodedNetSDF* representation consists of a neural network together with a latent code that encodes the specific particle morphology. To preserve object-level independence and modularity in the object-oriented design of *netdem*, different particle shapes represented by *CodedNetSDF* are stored as separate shape objects, each maintaining its own neural network parameters and the corresponding latent code. This design ensures that each particle shape is self-contained and can be independently managed, serialized, and loaded within large-scale DEM simulations. Alternative implementations in which neural network weights are shared across multiple particle instances are possible; however, the present design prioritizes object independence and straightforward integration within the DEM framework. By incorporating the NetSDF models, *netdem* enhances its ability to simulate complex granular materials, with the neural network providing an efficient means to represent particle shapes. This integration ensures that the DEM framework can handle a wide range of particle geometries, making it suitable for large-scale simulations of granular systems with varying particle shapes and interactions.

3. Model evaluation with shape reconstruction and generation

This section presents a series of numerical examples to verify and validate the effectiveness of the proposed NetSDF particle models. Both SingleNetSDF and CodedNetSDF are evaluated in terms of shape representation accuracy and their ability to generate virtual particle geometries.

3.1. Example particle templates and training dataset

As a first step toward evaluating the shape representation capabilities of the proposed NetSDF models, we employ a family of parametric superquadrics, specifically super-ellipsoids, as controlled test shapes. The super-ellipsoid is a closed superquadric surface defined by Barr (1981):

$$\left|\frac{x}{a}\right|^{\frac{2}{\epsilon}} + \left|\frac{y}{b}\right|^{\frac{2}{\epsilon}} + \left|\frac{z}{c}\right|^{\frac{2}{\epsilon}} = 1, \tag{2}$$

where a, b, and c are the semi-axes along the coordinate directions, and ϵ controls the edge roundness. By varying ϵ, this formulation can generate a wide range of particle geometries: as $\epsilon \to 0$, the shape approaches a cube; $\epsilon = 1$ yields a spheroid; and larger values lead to octahedral-like forms. This parametric flexibility makes the super-ellipsoid an ideal choice for testing how well NetSDF captures smooth, sharp-edged, and transitional shapes.

Based on this formulation, a set of 125 super-ellipsoidal particles was constructed by systematically varying the eccentricity index (EI), flattening index (FI), and shape exponent. These shapes serve as a structured benchmark for assessing the geometric accuracy and generalization performance of SingleNetSDF and CodedNetSDF. Representative examples are shown in Fig. 8.

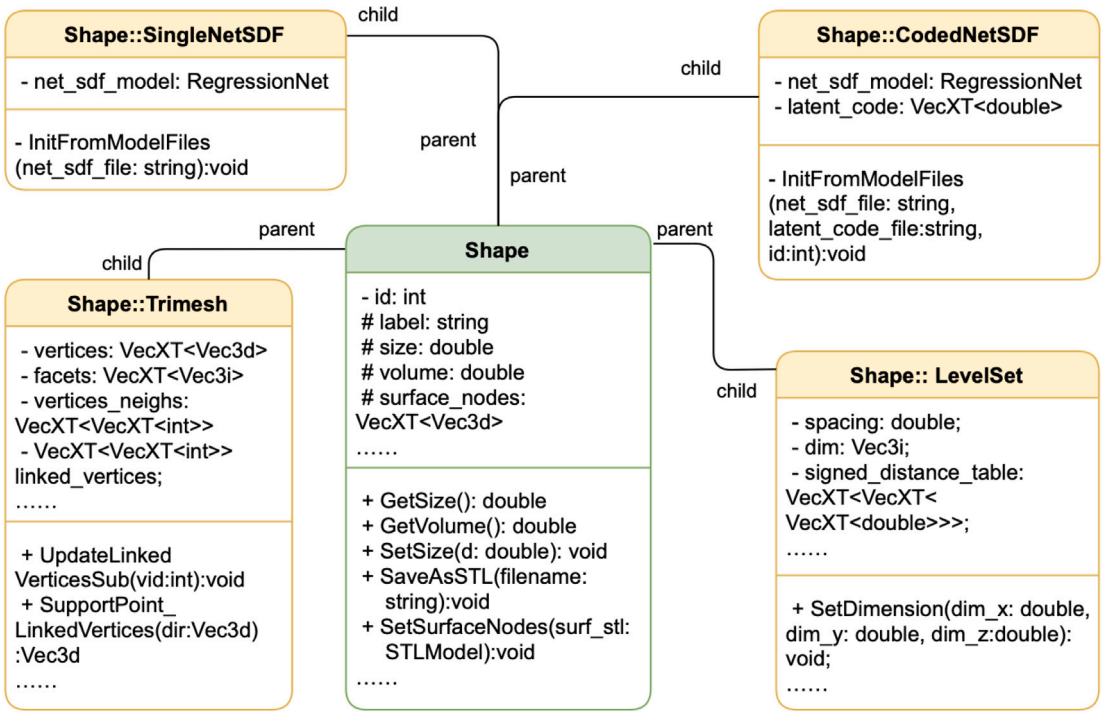


Fig. 7. Schematic diagram of the inheritance relationship of shape classes in *netdem*.

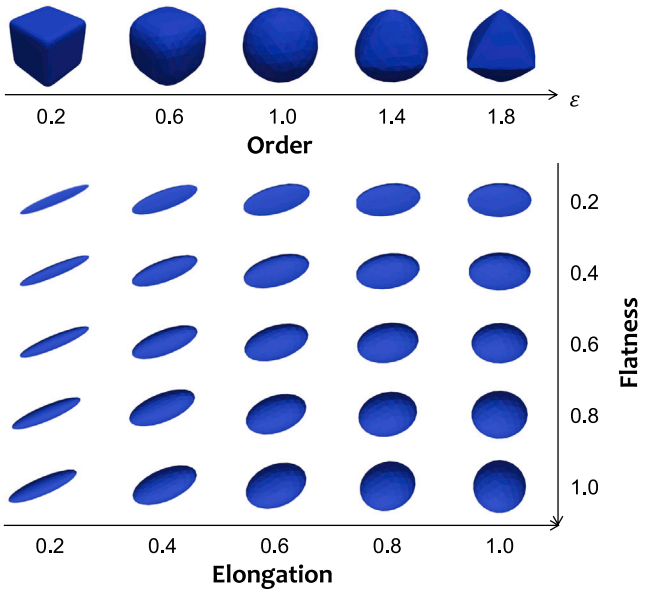


Fig. 8. Example super-ellipsoidal particles with varying shape parameters.

In addition to the super-ellipsoidal shapes, 20 complex particle geometries were selected to further evaluate the generalization capability of the proposed NetSDF models. These shapes are grouped into four categories (i.e., ballast, cobble, ellipsoid, sand), each containing five samples, and are designed to reflect practical use cases in transportation, geotechnical engineering, and pharmaceutical manufacturing. As shown in Fig. 9, the selected shapes feature irregular boundaries and varying topologies, providing a comprehensive test of the model's ability to capture diverse geometries encountered in real-world applications.

For all shape templates used in this study, training datasets were generated by uniformly sampling query points from a region bounded by two spheres: an expanded bounding sphere and a contracted inscribed sphere. Specifically, the outer and inner radii were defined by scaling the minimum bounding sphere by a factor of 1.1 and the maximum inscribed sphere by a factor of 0.9, respectively. This sampling strategy ensures that the training data sufficiently cover the spatial domain relevant to contact detection in DEM. Ground-truth signed distance values were computed using analytical SDF expressions, as detailed in Lai et al. (2022b). Unless otherwise specified, 10,000 training points were used per particle. The influence of training dataset size on NetSDF accuracy will be examined in the discussion section.

3.2. Results of SingleNetSDF for shape representation

Model training was conducted using the Adam optimizer with a learning rate of 0.001, a batch size of 128, and a maximum of 100 epochs. To ensure optimal generalization performance and explicitly mitigate the risk of overfitting, a quantitative 'best checkpoint' strategy was employed. Throughout the training process, the Mean Absolute Error (MAE) on the test set was continuously monitored alongside the training loss. While the training MAE typically decreases monotonically, the test MAE is prone to fluctuations or slight rebounds in later stages due to training noise. Therefore, rather than arbitrarily relying on the final epoch, the model snapshot corresponding to the global minimum of the test MAE was automatically identified and retrieved as the final output. This objective criterion enhances robustness by filtering out late-stage divergence. The efficacy of this strategy is illustrated in Fig. 10, which plots the evolution of both error metrics for the selected shape categories. As observed, specific minima (e.g., epoch 74 for Ballast and epoch 92 for Sand) are explicitly marked to denote the optimal stopping points, thereby achieving an optimal balance between convergence speed and generalization accuracy.

The practicality and robustness of the SingleNetSDF model were evaluated across the full set of 125 parametric shapes, spanning a wide range of elongation, flatness, and order parameters. Each particle was assigned an individual neural network for SDF prediction, and the performance (in terms of MAE) is summarized in Fig. 11. Each subfigure corresponds to a specific order value (ranging from 0.2 to 1.8), with

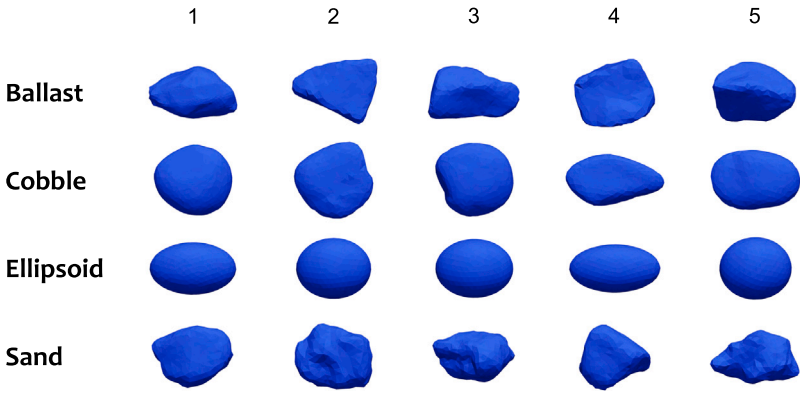


Fig. 9. Example complex geotechnical particle shapes.

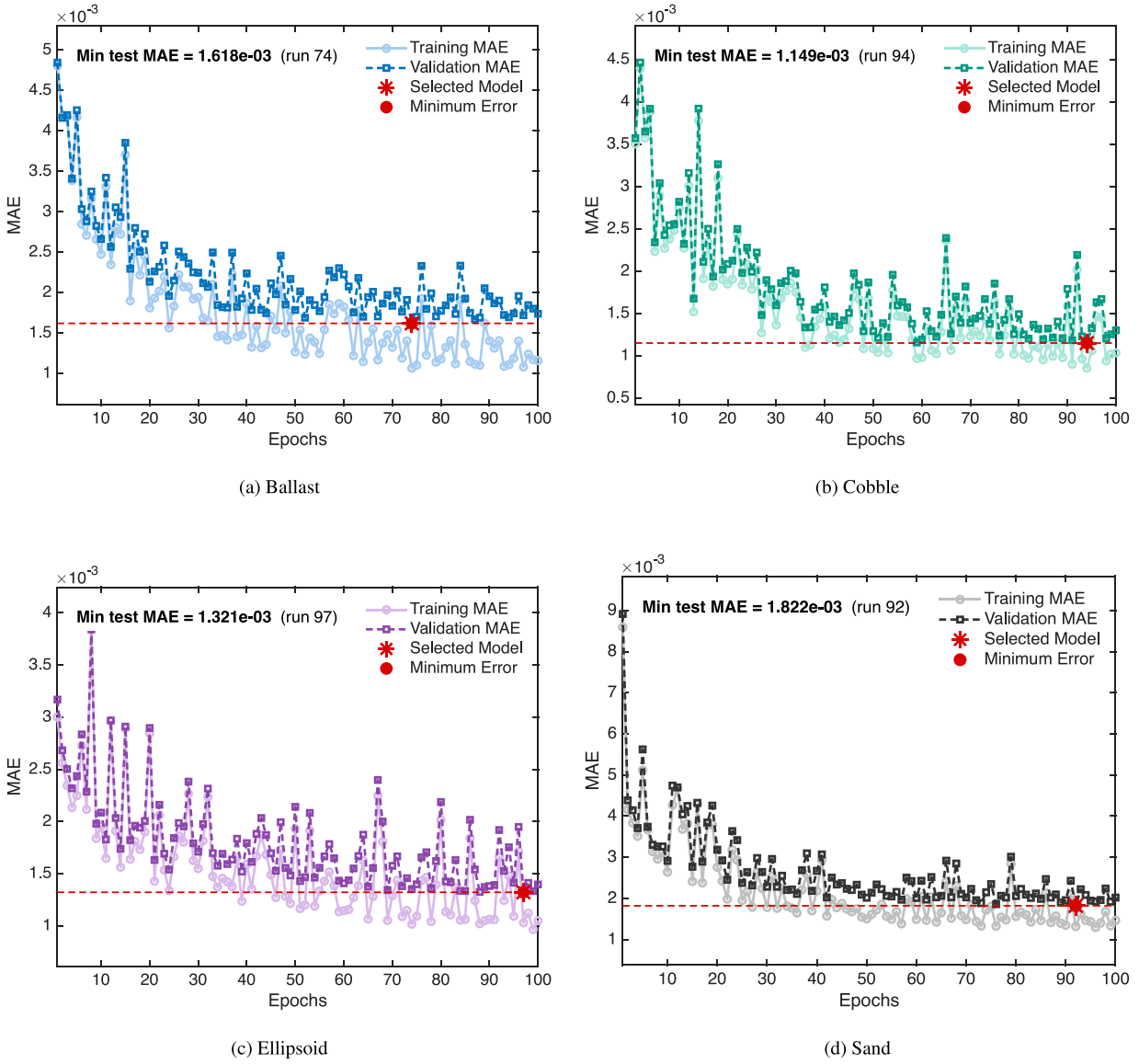
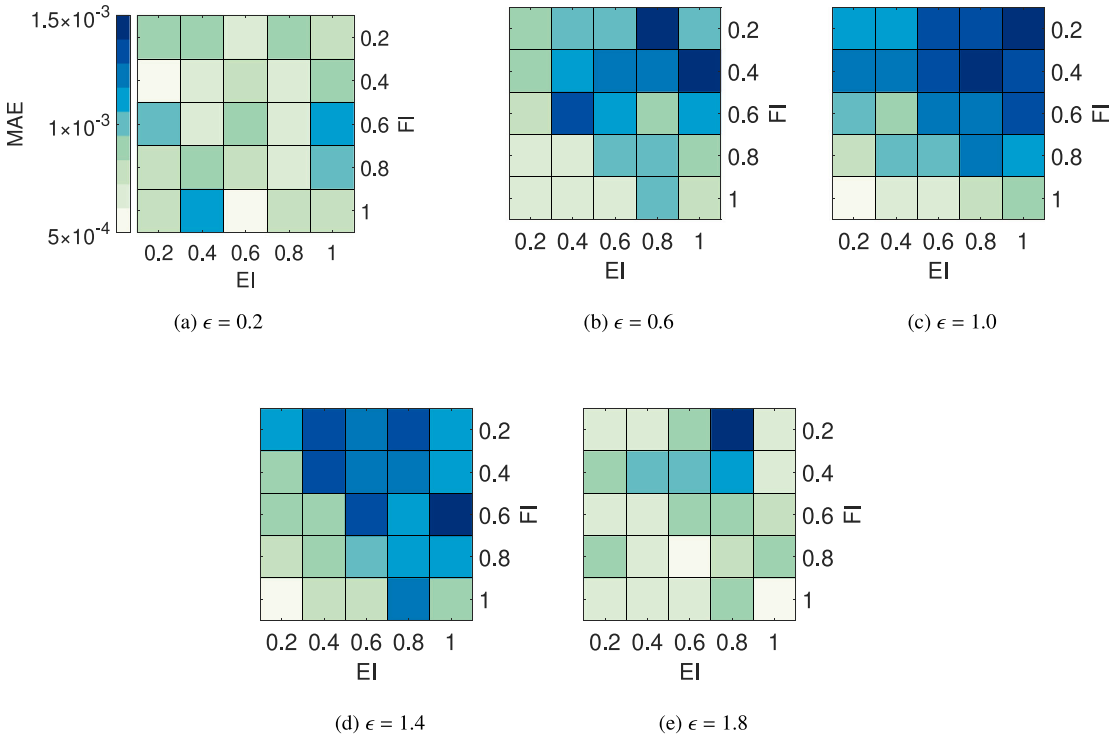


Fig. 10. Evolution of training and testing MAE for four representative particle shapes.

the heatmaps visualizing the MAE over the EI-FI space. Darker shades indicate higher prediction errors, while lighter tones signify higher fidelity. The results show that SingleNetSDF maintains low prediction error across all configurations, including highly elongated or flattened particles. This demonstrates the model's robustness and generalization capacity across a diverse morphological spectrum.

To further validate the model's performance on realistic geometries, we assessed the SingleNetSDF model on four representative natural



(a) $\epsilon = 0.2$ (b) $\epsilon = 0.6$ (c) $\epsilon = 1.0$

(d) $\epsilon = 1.4$ (e) $\epsilon = 1.8$

Fig. 11. MAE of SDF prediction across 125 parametric shapes using SingleNetSDF. Each heatmap shows the error distribution over elongation (EI) and flatness (FI) for a fixed order value; darker shades indicate larger errors.

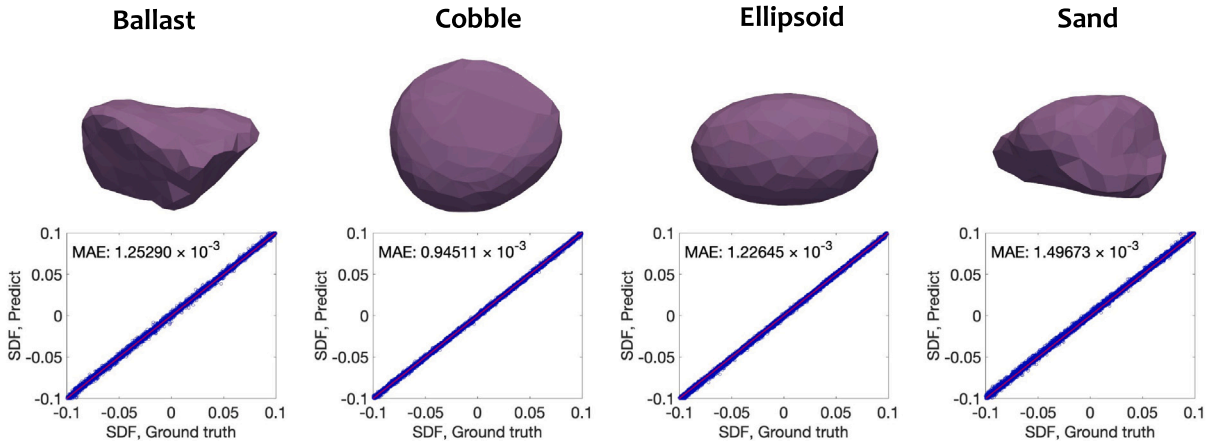


Fig. 12. Comparison between predicted and ground truth particle shapes using SingleNetSDF, including 3D surface reconstructions (top) and SDF value correlations (bottom).

particle shapes: ballast, cobble, ellipsoid, and sand. As shown in Fig. 12, the reconstructed 3D surfaces (top row) closely match the ground truth models, accurately capturing both smooth contours and complex irregularities. The corresponding scatter plots (bottom row) compare predicted and reference SDF values, exhibiting tight alignment along the diagonal and consistent MAEs on the order of 10^{-3}. These results demonstrate the model's ability to handle real-world shape complexity and reinforce its applicability in DEM simulations requiring precise geometric fidelity.

3.3. Results of CodedNetSDF for shape representation

The performance of the CodedNetSDF model was assessed in terms of convergence behavior, reconstruction accuracy, and scalability in encoding multiple shapes with shared network parameters. To begin with, CodedNetSDF was trained using grouped particle shapes from the earlier dataset, where each group was encoded with its own set of latent codes. The evolution of test accuracy over training epochs is illustrated in Fig. 13. The network converges rapidly within the first 100 epochs, particularly for smoother geometries, before reaching a plateau near 200 epochs. While shapes with regular surfaces exhibit faster and more stable convergence, more irregular geometries lead to slower learning and higher final errors. This behavior reflects the increased difficulty in optimizing latent representations for complex morphologies.

To rigorously evaluate the model's robustness against geometric anisotropy, two distinct particle templates were intentionally selected from the super-ellipsoid dataset to represent morphological extremes: a near-perfect sphere (acting as an isotropic baseline) and a highly elongated, strip-like particle (representing the anisotropic extreme). As shown in Fig. 14, the CodedNetSDF model achieves uniform high-fidelity reconstruction for both geometries. This comparison is critical because highly anisotropic shapes often impose greater demands on

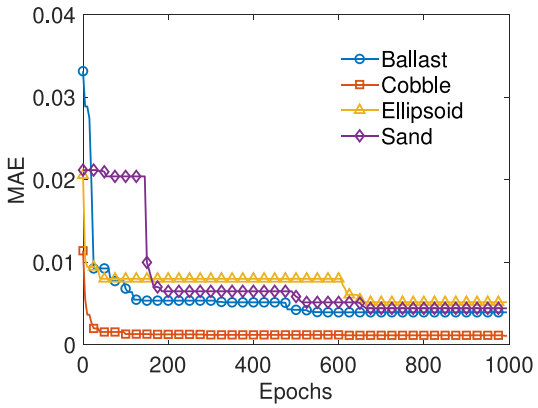


Fig. 13. Convergence behavior of the CodedNetSDF model during training. Each curve represents the test accuracy trajectory for a representative particle shape.

the latent space representation, leading to potential encoding errors. By maintaining a reconstruction error for the strip-like particle that is comparable to that of the simple sphere, the model demonstrates that its learned latent manifold is robust and capable of accurately capturing complex curvatures regardless of significant variations in aspect ratio.

Moving beyond these parametric extremes to practical application, we extended the evaluation to realistic granular materials to verify the model's capability in capturing complex, irregular morphologies. Specifically, the shared CodedNetSDF network was employed to reconstruct the four representative particle types defined earlier: ballast, cobble, ellipsoid, and sand. As visually presented in Fig. 15, the reconstructed 3D surfaces (top row) exhibit exceptional visual fidelity compared to the ground truth meshes, successfully preserving intricate features such as the sharp angular edges of ballast and the smooth curvature of cobbles. Quantitatively, the corresponding scatter plots (bottom row) corroborate this observation, revealing a strong correlation between the predicted and ground truth SDF values. With MAEs consistently remaining at the order of 10^{-3}, these results confirm that the CodedNetSDF framework effectively encodes diverse realistic geometries within a unified latent space without significant loss of precision.

To evaluate the scalability of the CodedNetSDF framework, a single network was trained to represent 20 particle shapes simultaneously, with each assigned a unique 10-dimensional latent code. The shapes span a diverse range of morphologies from the earlier dataset. The reconstruction errors are summarized in Fig. 16, where rows correspond to particle categories and columns index individual shapes. MAE values predominantly fall between 0.01 and 0.05, indicating good overall performance. However, noticeable differences emerge: smoother shapes are reconstructed with lower error, while more angular or irregular ones yield higher MAEs, occasionally exceeding 0.04. These results demonstrate the trade-off between latent space compactness and shape fidelity, and suggest that shape complexity plays a key role in determining encoding effectiveness.

These findings confirm the feasibility of using CodedNetSDF for efficient multi-shape representation within a shared latent space. Nonetheless, they also reveal challenges in latent code optimization for irregular geometries. The effects of model architecture, such as latent code dimensionality, neural network capacity, and grid resolution, on reconstruction performance will be discussed in subsequent sections.

3.4. NetSDF for virtual shape generation

Beyond accurate shape reconstruction, CodedNetSDF offers a generative capability by leveraging the structured latent space of a trained network to synthesize novel particle geometries. One intuitive and powerful approach is linear interpolation between latent codes, which enables smooth and continuous transitions between two reference particles. This method facilitates controlled morphological variations within a particle category while preserving essential shape characteristics. Representative examples are presented in Fig. 17, where sand and ballast particles exhibit gradual and coherent transitions across the latent space. As evident from the interpolated intermediate shapes, the model captures not only global geometric trends, such as elongation and roundness, but also local features such as surface roughness and angularity. The visual smoothness of these transitions highlights the continuity and expressiveness of the learned latent manifold.

In addition to within-category interpolation, CodedNetSDF supports cross-category transitions, as shown in Fig. 18. This figure demonstrates interpolated sequences connecting four distinct particle types, including sand, cobble, ballast, and ellipsoid. Despite their differing origins and structural features, the model is able to produce intermediate shapes that remain geometrically coherent and physically plausible. Such capability underscores the generality and flexibility of the NetSDF framework in capturing a broad morphological spectrum within a unified representation. This paves the way for virtual exploration of hybrid geometries or design of functionally graded particle ensembles without requiring network retraining.

Apart from interpolation, CodedNetSDF also enables the stochastic generation of new shapes by sampling from the latent distribution learned during training. As illustrated in Fig. 19, random latent codes were drawn from a multivariate Gaussian distribution estimated over existing shape categories. For visualization, the high-dimensional (10D) latent vectors are projected into 2D space using the t-distributed stochastic neighbor embedding (t-SNE) algorithm (Maaten and Hinton, 2008). The spatial layout in Fig. 19(a) reveals meaningful clustering patterns, while the corresponding samples in Fig. 19(b) show that the synthesized shapes are diverse yet structurally realistic. Notably, these generated particles exhibit non-trivial geometric features such as asymmetric lobes, varying curvature, and localized sharpness, which are difficult to obtain through procedural modeling.

Physically, this sampling process can be interpreted as exploring the statistical manifold of valid particle geometries. The multivariate Gaussian distribution, fitted to the latent codes of the training particles, captures the inherent correlations between geometric features. Consequently, the sampling variance serves as a control parameter that balances shape realism against morphological diversity. Sampling within the high-probability region (e.g., near the mean) yields particles that strictly adhere to the prototypical features of the category, ensuring high physical plausibility. Conversely, increasing the sampling variance yields 'extreme' geometries with exaggerated morphological features, such as higher aspect ratios and sharper edges. While these outliers are valuable for stress-testing DEM simulations and analyzing limiting cases of contact behavior, excessive variance generally increases the risk of geometric artifacts or degenerate shapes. To mitigate this risk in practical DEM applications, specific constraints on the sampling strategy are recommended. A primary safeguard is truncated sampling, where the random generation is confined to a bounded region (e.g., within ± 2 standard deviations of the cluster centers). This strategy effectively prevents the synthesis of unrealistic outliers while preserving sufficient morphological variation consistent with the training dataset. Furthermore, for applications with strict geometric requirements, a descriptor-based screening process (i.e., rejection sampling) can be employed as a robust post-processing step. By filtering generated candidates based on key physical descriptors, such as sphericity or convexity indices, users can ensure that all initialized particles strictly satisfy the domain-specific criteria for physical fidelity.

Overall, validation using known particle categories confirms that CodedNetSDF can reliably interpolate both within and across shape families, while also supporting diverse shape synthesis through latent code sampling. These capabilities position the framework not only as

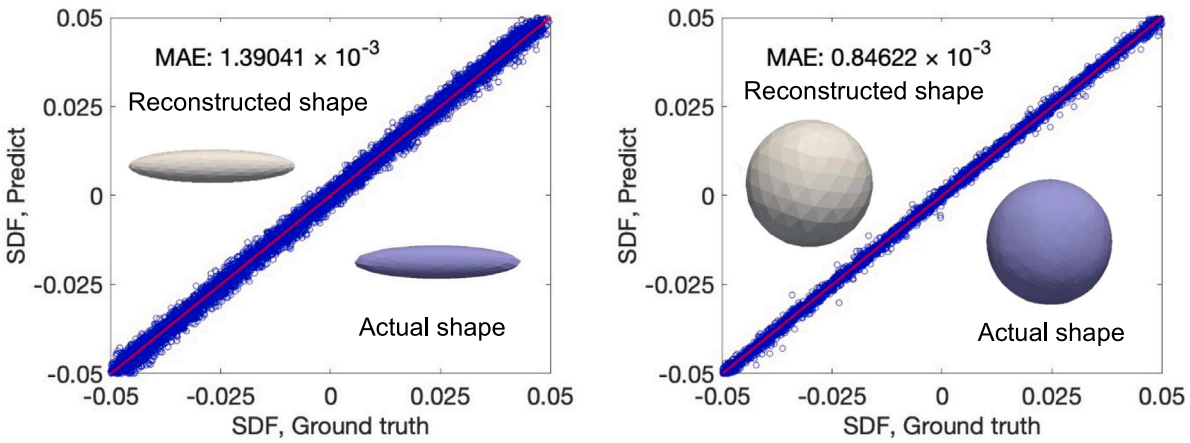


Fig. 14. Comparison of SDF prediction and ground truth for two representative shapes using CodedNetSDF.

a compact and accurate shape representation model, but also as a generative engine for virtual particle design. The ability to explore morphological variations-whether for DEM simulations, material optimization, or data-driven design tasks-offers a practical and scalable approach to integrating learned shape priors into performance-oriented granular modeling workflows.

4. Model parameter analysis and computational performance

This section presents a detailed investigation of the factors that affect the accuracy, generalization, and computational performance of the NetSDF framework. Specifically, we examine how training data resolution, neural network architecture, and latent space dimensionality influence model behavior. In addition, computational efficiency metrics, including memory footprint and inference speed, are evaluated to assess the feasibility of integrating NetSDF into large-scale DEM simulations.

4.1. Effect of the number of training samples

The number of sampling points per particle directly influences the model's ability to capture surface details and reconstruct accurate SDFs. Fig. 20 shows the influence of training sample size on model accuracy. As expected, increasing the number of samples generally improves accuracy, although the rate of improvement is shape-dependent. However, the rate of improvement is shape-dependent. Ballast and sand particles show higher sensitivity to sampling density, with test error fluctuating when sample sizes are below 5000. This can be attributed to their geometrical irregularities and sharper surface features, which require denser sampling to adequately characterize the SDF field. In contrast, cobble and ellipsoid particles, which are featured with smoother and more regular surfaces, exhibit relatively stable error curves even under lower sampling resolutions. Beyond approximately 5000 sample points per shape, the accuracy gains begin to plateau, indicating diminishing returns from further increasing the sample count. This observation suggests that an optimal trade-off exists between sampling resolution and computational efficiency. In practice, 5000–10,000 sample points per shape provide a good balance for most geometries considered in this study.

It is also worth noting that the sampling strategy plays a role in how efficiently these geometric features are captured. The current study adopts a uniform sampling method within a 'narrow band' (bounded by 0.9 and 1.1 times the inscribed/circumscribed sphere radii). This spatial restriction ensures that training points are concentrated in the contact-relevant zone, which explains why the model achieves high fidelity (MAE $\approx 10^{-3}$) even for angular particles like ballast, as shown in Fig. 12. However, as indicated by the slower convergence of ballast in Fig. 20, uniform sampling may be suboptimal for strictly capturing sharp edges or high-curvature regions. To better capture sharp geometric features, future implementations could adopt curvature-aware sampling, where higher point densities are allocated to high-curvature regions. Our current uniform 'narrow-band' method provided sufficient fidelity for the tested materials, but adaptive sampling offers a clear path toward further optimization.

4.2. Effects of network configuration

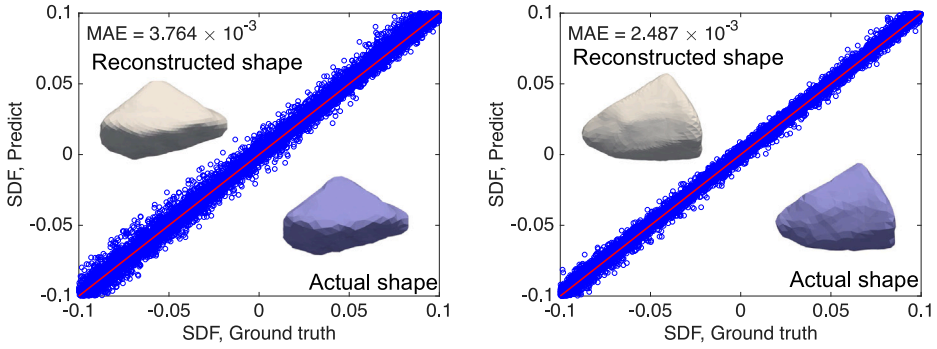
The architecture of the neural network, including the number of hidden layers and the number of nodes per layer, plays a critical role in balancing model expressiveness and generalization. This subsection investigates how these architectural parameters affect the prediction accuracy of both SingleNetSDF and CodedNetSDF models.

4.2.1. Number of layers

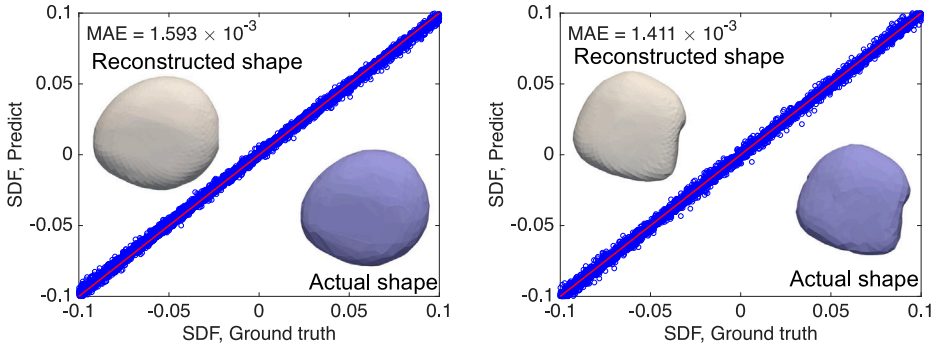
Fig. 21 illustrates the effect of increasing the number of hidden layers on the prediction accuracy, quantified by MAE. For the SingleNetSDF model (Fig. 21(a)), the MAE exhibits a noticeable drop as the number of layers increases from 2 to 3, especially for ballast and sand particles, indicating a clear gain in accuracy. However, beyond 4 layers, the error begins to fluctuate, and further increasing the depth to 9 layers does not yield consistent improvements. In fact, for sand particles, deeper networks tend to degrade performance, possibly due to overfitting or increased training instability. For the CodedNetSDF model (Fig. 21(b)), a similar pattern is observed. The MAE decreases sharply when moving from 1 to 3 layers, especially for ballast and sand, but plateaus or slightly increases beyond 5 layers. Among all particle types, ellipsoids consistently achieve the lowest error, reflecting their geometric simplicity and smoothness, while sand particles remain the most sensitive to layer depth due to their irregular morphology. Overall, these results suggest that an optimal number of hidden layers exists, around 4–5 layers for CodedNetSDF and 5–6 layers for SingleNetSDF regarding the studied shapes, that balances model expressiveness with generalization ability. Adding more layers beyond this range provides marginal or even detrimental effects on prediction accuracy.

4.2.2. Number of nodes per layer

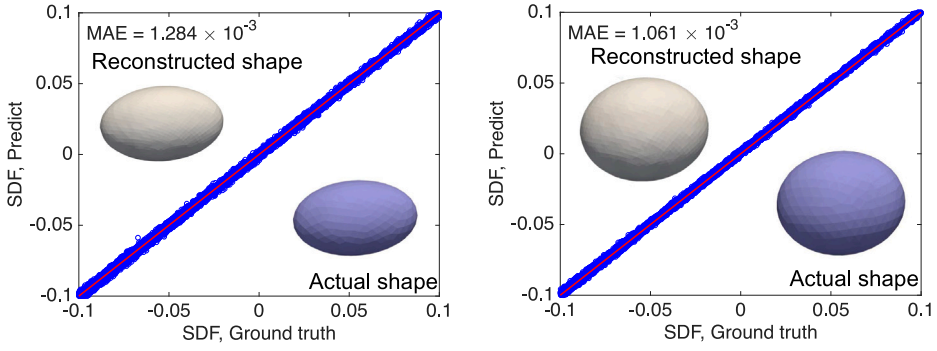
Fig. 22 shows the impact of varying the number of nodes in each hidden layer. For SingleNetSDF (Fig. 22(a)), a substantial improvement in accuracy is observed as the node count increases from 15 to 30, particularly for ballast particles. This improvement tapers off beyond 40 nodes, where only marginal gains are seen, especially for cobble and ellipsoid shapes. A similar pattern is evident in the CodedNetSDF model (Fig. 22(b)). Increasing the number of nodes from 20 to 50 significantly improves prediction accuracy, with the most pronounced effect seen in



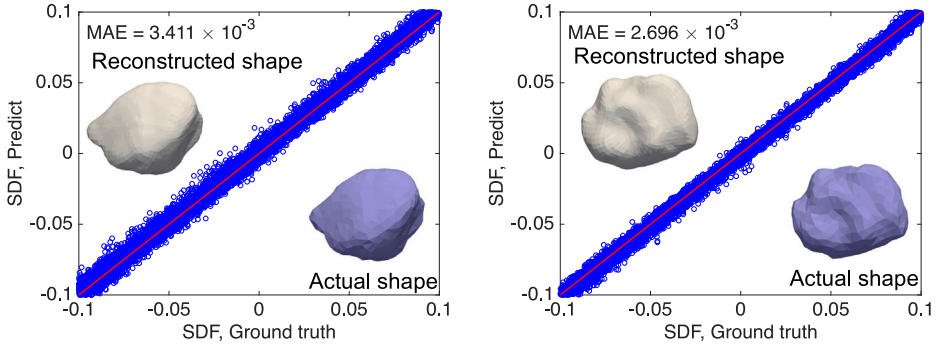
(a) Ballast



(b) Cobble



(c) Ellipsoid



(d) Sand

Fig. 15. Comparison of SDF prediction and ground truth for four representative particle shapes using CodedNetSDF.

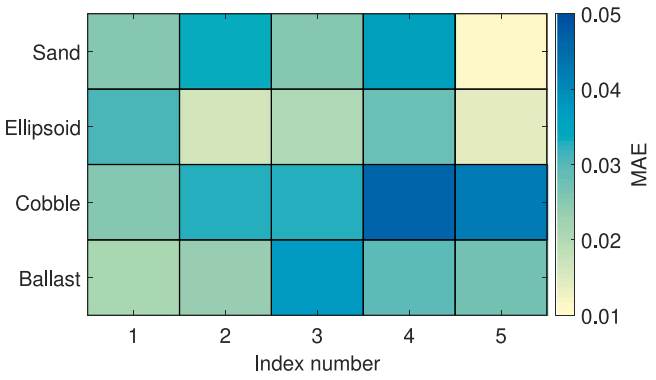


Fig. 16. Reconstruction error (MAE) of CodedNetSDF trained on 20 shapes using shared weights. Rows denote shape categories; columns index individual shapes.

ballast particles. However, expanding the node count beyond 70 yields negligible improvements. These results suggest an optimal range of 30–50 nodes per layer for most particle types, offering a good trade-off between network capacity and computational efficiency.

4.3. Effect of latent dimensionality

The latent dimensionality plays a crucial role in balancing model compactness and shape representation fidelity. Fig. 23 reports the MAE as a function of latent dimensionality for four particle categories. Across all configurations, ellipsoid particles consistently achieve the lowest MAE, underscoring that their smooth and regular geometry can be effectively captured even with low-dimensional embeddings. Cobble particles also exhibit stable reconstruction performance, with MAE values fluctuating mildly and showing limited sensitivity to latent dimensionality. In contrast, ballast and sand particles, both characterized by more irregular surface features, show greater MAE variability across different latent dimensionality. For ballast, the MAE initially decreases with longer codes, reaching a minimum around dimension 10 before fluctuating in the range of 3.0×10^{-3} to 4.0×10^{-3}. Sand particles exhibit a similar trend but with higher overall error and less pronounced convergence. These results suggest that geometrically complex particles may benefit from moderately larger latent spaces to adequately encode intricate shape details, although gains tend to plateau beyond a certain dimensionality. Overall, latent dimensionality between 8 and 12 offers a favorable trade-off for most particle types, providing sufficient representational capacity without introducing redundant or noisy features. This balance is essential for ensuring both accuracy and generalization when deploying learned particle shapes in downstream simulation or generative tasks.

4.4. Computational performance

4.4.1. Memory usage

In large-scale DEM simulations, memory consumption per particle becomes a critical bottleneck. As the number of particles increases to improve the physical realism of the simulation, the total memory required to store their geometries scales accordingly. Efficient particle representation methods must therefore minimize memory overhead while preserving shape fidelity. Table 1 compares the average memory usage per particle for several shape representation strategies, based on a test case involving 100,000 particles. The conventional triangular mesh (Trimesh) approach demands the most memory, averaging approximately 387 kB per particle due to its explicit surface discretization. The level set method offers tunable memory cost depending on the resolution: at a resolution of 0.033 (i.e., with grid size being 1/33), memory usage exceeds 600 kB, while a coarser resolution (grid size 0.050, i.e., 1/20) reduces it to around 235 kB. In contrast, the neural representation approaches, namely SingleNetSDF and CodedNetSDF, achieve substantially better memory efficiency. SingleNetSDF requires only 48.4 kB per particle, while CodedNetSDF uses about 66.1 kB, the latter including a compact latent code vector for shape encoding. These results highlight the scalability of the proposed methods, especially for DEM simulations involving large number of particles or limited computational resources.

Table 1
Memory usage (per particle) of different shape representation methods.

Method	Memory Usage (kB)
Triangular Mesh (Trimesh)	386.99
Level Set (resolution: 0.050)	235.11
Level Set (resolution: 0.033)	615.25
SingleNetSDF[a]	48.42
CodedNetSDF[b]	66.12

[a] SingleNetSDF stores a single embedded neural SDF model without a latent code.
[b] CodedNetSDF stores an embedded neural SDF model together with a particle-specific latent code and minimal metadata.

4.4.2. Computation time

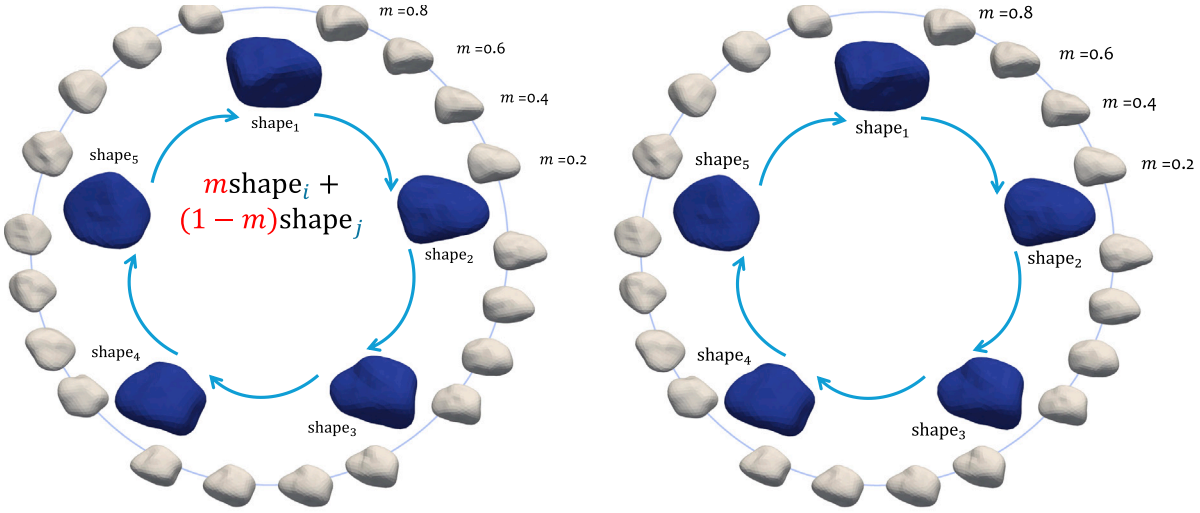
In the SDF-DEM, contact detection and interaction forces are typically computed by querying the surface nodes of one particle with respect to another. Therefore, evaluating the speed at which these queries can be processed is vital for assessing the computational efficiency of different shape representation methods. To quantify time performance, one million spatial queries were randomly distributed within a unit bounding box encompassing a single particle, and the total evaluation time was recorded. As shown in Fig. 24, the level set method exhibits the fastest response time, approximately two orders of magnitude faster than the other methods, due to its structured grid and efficient distance interpolation scheme. Among learning-based representations, SingleNetSDF and CodedNetSDF demonstrate competitive speed, with both reducing computation time by nearly 50% compared to the Trimesh baseline. This is primarily due to the amortized cost of neural inference and avoidance of complex mesh-based geometric checks. Although not as fast as level set methods, they offer a superior trade-off between speed, memory, and shape fidelity, making them particularly attractive for simulations where high geometric accuracy is required without incurring excessive computational cost.

5. Example applications via DEM simulation

To validate and demonstrate the performance of NetSDF, a series of representative simulations were conducted, including random stacking, rotating drum, funnel discharge, and slope rockfall. These simulations address classical problems in computational particle mechanics, exploring both dynamic and quasi-static conditions, thus providing a comprehensive evaluation of the applicability and robustness of the proposed methods.

5.1. Random packing

The first validation scenario involves a classical random packing simulation using discrete elements. Four particle types were selected from the previously trained multi-shape CodedNetSDF model: ballast, cobble, ellipsoid, and sand. To assess both single-type and mixed-type behavior, simulations were performed separately for each type and for a mixed assembly. All particles had an equivalent diameter of 0.1 m and a density of 2650 kg/m^3. The interactions were governed by a linear spring–dashpot contact law with the following parameters: normal stiffness 2×10^6 N/m, tangential stiffness 1×10^6 N/m, friction



(a) Sand (b) Ballast

Fig. 17. Shape interpolation within sand and ballast categories using CodedNetSDF.

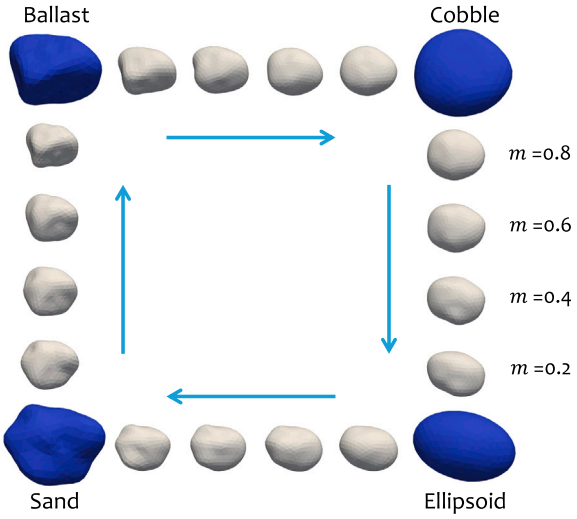


Fig. 18. Cross-category interpolation results using CodedNetSDF for four representative particle shapes.

coefficient 0.5, and damping coefficient 0.7. The particles were packed into a square container of size 1.0 m by 1.0 m under gravity-driven deposition.

All simulations exhibited excellent numerical stability, even in cases involving mixed particle morphologies. The SDF-based DEM solver robustly captured complex contact interactions without producing interpenetration artifacts or convergence issues. To facilitate visualization and further analysis, the simulated particle assemblies were exported as STL meshes and rendered using ParaView. Representative snapshots are presented in Fig. 25, illustrating the method's effectiveness in handling a wide range of particle geometries within a unified simulation framework.

5.2. Rotating drum

To further evaluate the numerical stability and dynamic performance of the proposed NetSDF and SDF-DEM framework, a rotating drum simulation was carried out. This setup is widely used to study granular flow behavior under continuous deformation and shear. The drum was modeled as a vertical cylinder with a diameter of 1.0 m and height of 0.3 m, rotating around its central axis at a constant angular velocity of -0.2π rad/s. Ballast-type particles, reconstructed using CodedNetSDF, were first randomly inserted into the drum and allowed to settle under gravity until a stable packing state was reached. The particles had an average diameter of 0.1 m and interacted via the same contact parameters used in the previous random packing test.

After the initial packing phase, rotation was initiated, and particle dynamics were monitored. Representative snapshots are shown in Fig. 26. As the drum rotated, the particles displayed continuous avalanching and formed a well-defined, stable angle of repose, i.e., a hallmark of granular flow under rotational loading. Throughout the simulation, the solver maintained excellent numerical stability: no spurious interpenetration, oscillations, or energy divergence were observed. These results validate the robustness and accuracy of the CodedNetSDF-integrated DEM approach in capturing realistic particle-scale behaviors under dynamic conditions involving complex contact evolution.

5.3. Funnel discharge

The third test focused on a large-scale funnel discharge simulation involving 12,000 particles to further evaluate the scalability and robustness of the CodedNetSDF-based DEM framework. The simulation mimics the flow and accumulation of granular materials, such as pills, seeds, or grains, through a funnel into a confined domain. All particles were represented by ballast geometries generated via CodedNetSDF. Each particle had a characteristic diameter of 0.1 m and interacted through the same contact parameters as previous tests.

Initially, particles were positioned above the funnel and released under gravity. As they passed through the narrow opening and settled, the simulation captured realistic flow characteristics, including arching effects, funnel blockage, and avalanching near the outlet. The final pile exhibited a well-defined and stable angle of repose. Snapshots illustrating the initial geometry, intermediate flow, and final accumulation are provided in Fig. 27. Throughout the simulation, the solver maintained strong numerical stability, with no observed interpenetration, spurious particle oscillations, or divergence in kinetic energy. This confirms the suitability of the proposed approach for high-particle-count simulations involving complex geometries and dynamic interactions.

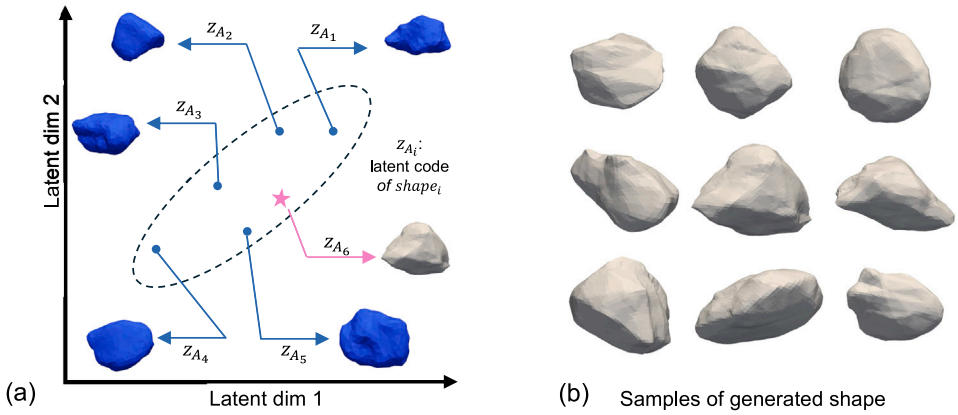


Fig. 19. Latent space sampling (a) and corresponding generated shapes (b) using multivariate Gaussian distribution.

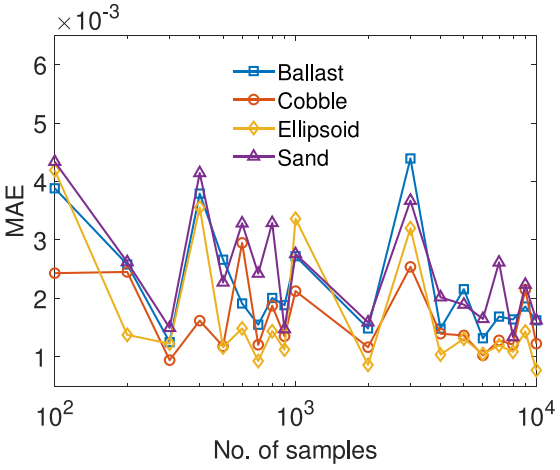


Fig. 20. Effect of the number of sampling points per particle on SDF prediction error (SingleNetSDF).

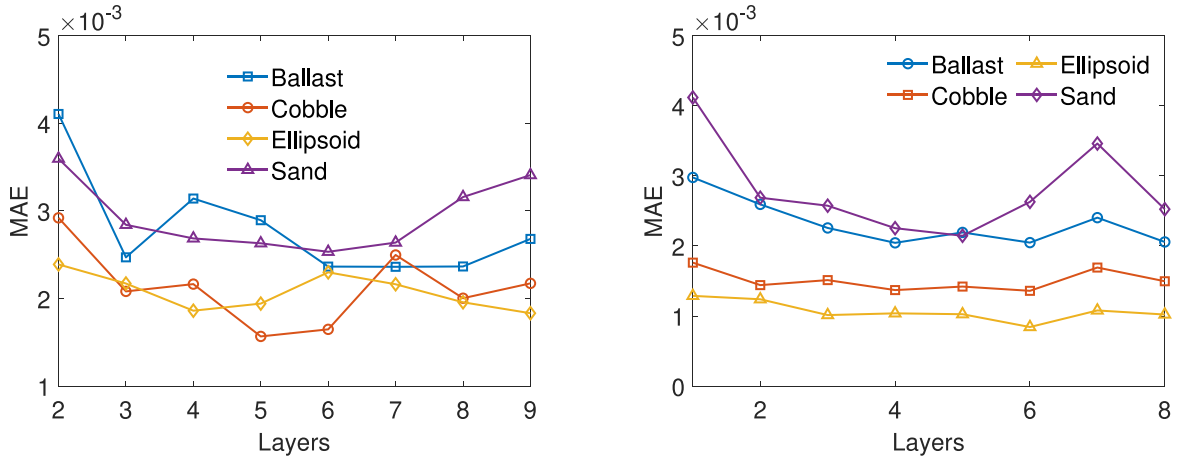
5.4. Slope rockfall

Existing studies have increasingly recognized the critical role of particle morphology in slope stability and rockfall trajectory prediction (Qin et al., 2025; Yoshida et al., 2024). Building upon this engineering context, we conducted a representative case study to demonstrate the applicability of the NetSDF framework in geohazard assessment. The simulation was performed on a realistic terrain model constructed from real-world elevation data and imported as a triangular mesh (STL), featuring complex uneven surfaces that pose challenges for robust contact detection. The primary objective was to evaluate how subtle morphological variations influence the kinematic behavior and trajectory of falling rocks. Leveraging the generative capability of CodedNetSDF, we established a diverse particle ensemble comprising three distinct categories: Scanned Reference shapes reconstructed directly from scanned ballast, Latent Interpolation shapes obtained via linear transition of latent codes, and Stochastic Sampling shapes randomly drawn from the learned multivariate distribution. All particles were initialized at identical release points along the slope crest with a vertical initial velocity of −1.0 m/s and interacted with the terrain using the same contact parameters as previous validation tests. This setup effectively isolates the influence of particle geometry, allowing for a rigorous examination of how the generated shape library contributes to trajectory prediction.

The simulation results, summarized in Fig. 28, highlight the significant divergence in kinematic behavior driven by morphological diversity. Specifically, the visual comparison distinguishes between the Scanned Reference (green), Latent Interpolation (blue), and Stochastic Sampling (red) particles. As illustrated in the perspective and top-down views, despite identical initial conditions and material properties, the particles follow distinct spatial trajectories with pronounced lateral dispersion. This phenomenon arises from the unique shape-dependent contact dynamics, such as irregular bouncing and rolling, induced by specific angular features interacting with the terrain. Consequently, the simulation underscores that the morphological variability captured by the latent space plays a critical role in determining both runout paths and spatial spread.

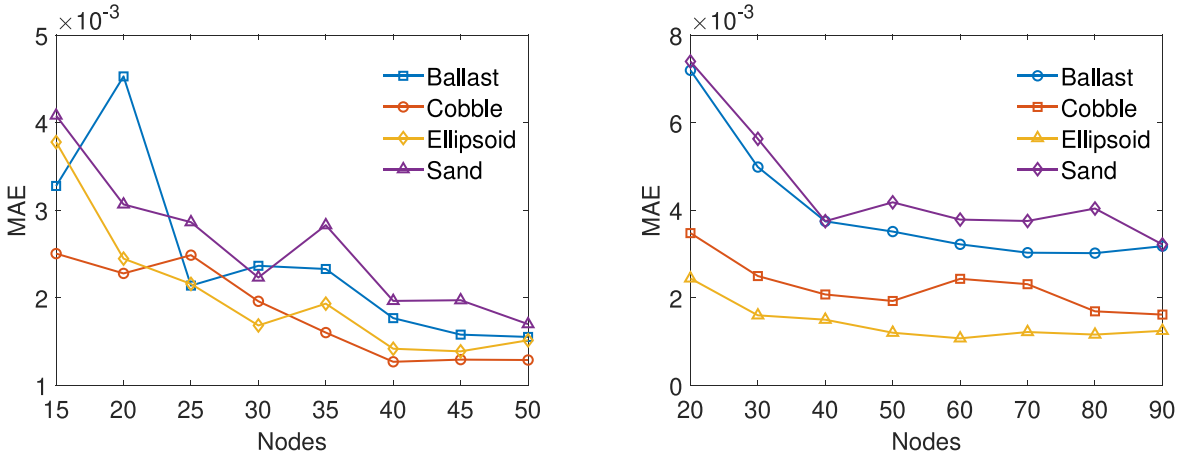
This kinematic variability is further quantified in Fig. 29, which presents a matrix of velocity magnitude time-histories for three representative cases within each category. The curves reveal a wide spectrum of motion patterns, characterized by significant variations in peak velocities, collision frequencies, and runout durations. While certain particles rapidly dissipate energy and settle in local depressions (e.g., Scanned Reference Case I), others exhibit high-velocity bounds and prolonged movement (e.g., Latent Interpolation Case I and Stochastic Sampling Case II). These findings demonstrate that simulations relying on a single 'representative' shape may fail to capture the full envelope of potential kinetic energy and impact zones. Therefore, the CodedNetSDF framework serves as a valuable tool for safety assessment, enabling engineers to efficiently generate and simulate a diverse library of realistic shapes to identify potential risk scenarios with greater confidence.

Overall, these simulation cases, including random stacking, rotating drum, funnel discharge, and slope rockfall, demonstrate the versatility and robustness of the NetSDF framework across a range of classical problems in computational particle mechanics. These examples encompass both quasi-static and dynamic conditions, highlighting the framework's ability to model shape-dependent behaviors under diverse loading regimes. It should be emphasized that the particle numbers in these simulations were chosen primarily for validation purposes rather than to reflect the computational upper limit of the solver. The underlying NetDEM framework, built upon MPI and OpenMP parallelism, is architecturally capable of handling simulations with millions of particles (Lai et al., 2022b). However, scaling neural network inference to such magnitudes introduces a computational bottleneck, as the current implementation relies on the CPU-based mlpack library. Given the feed-forward structure of the network, which is inherently well-suited to single-instruction-multiple-thread (SIMT) parallelism, future work will focus on transitioning the inference backend to GPU-accelerated frameworks such as CUDA or LibTorch. This enhancement is expected to enable high-throughput batched inference of spatial queries, thereby overcoming the primary limitation in applying neural implicit models to industrial-scale granular simulations.



(a) SingleNetSDF (b) CodedNetSDF

Fig. 21. Effect of the number of hidden layers on prediction accuracy (measured by MAE).



(a) SingleNetSDF (b) CodedNetSDF

Fig. 22. Effect of the number of nodes per layer on SDF prediction error (MAE).

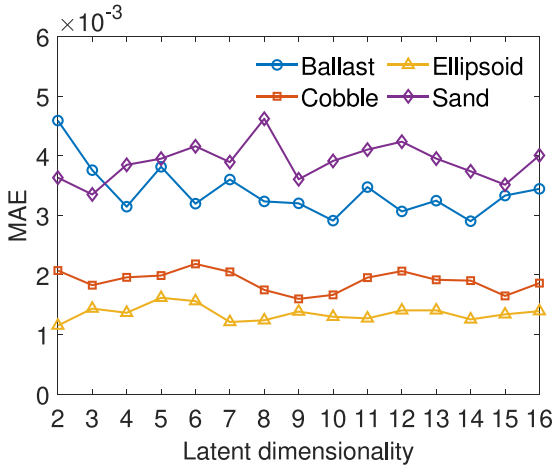


Fig. 23. Effect of latent dimensionality on SDF reconstruction error (MAE) for a single particle class in CodedNetSDF.

6. Summary

This study presents NetSDF, a neural implicit framework for shape representation and granular particle simulation based on signed distance fields (SDFs). To accommodate distinct engineering requirements, two complementary model variants are explored: SingleNetSDF, which is efficiency-oriented and optimizes the reconstruction of specific, fixed geometries (e.g., Digital Twins) by learning a separate, compact neural network for each particle; and CodedNetSDF, which is functionality-oriented, utilizing a shared network with low-dimensional latent codes to enable statistical shape analysis and stochastic generation. These models are integrated into an SDF-based discrete element method (SDF-DEM), enabling contact-resolving simulations with arbitrary particle geometries. To support physical interaction and post-processing, the framework also introduces two surface projection algorithms, gradient descent and ray-searching, for accurate contact point recovery, along with mesh reconstruction techniques based on alpha shapes and spherical harmonics. Together, these components form a unified pipeline for shape-aware particle modeling, simulation, and generation.

Extensive numerical experiments validate the accuracy, efficiency, and applicability of the proposed approach. On a structured dataset of 125 parametric super-ellipsoids and 20 realistic particle shapes, both model variants achieve mean absolute SDF errors below 10^{-3} (with respect to unit particle size), with stable convergence across a range

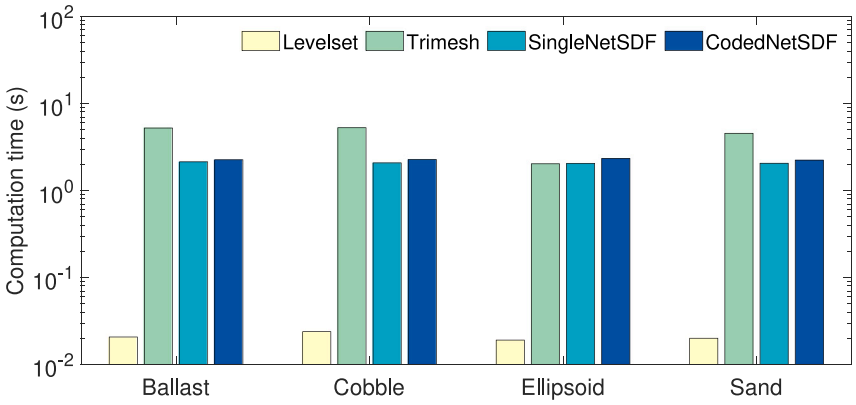


Fig. 24. Computation time (log scale) for different shape representations based on one million spatial queries per particle.

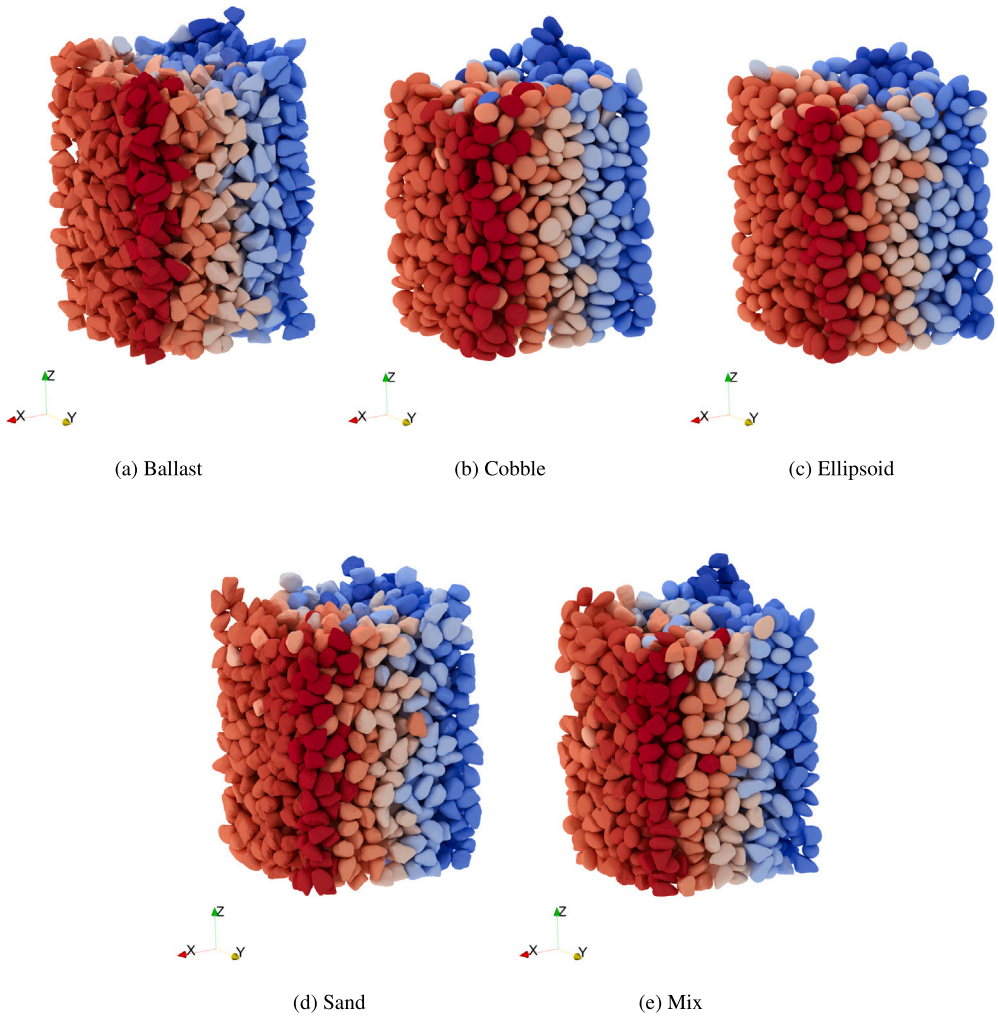
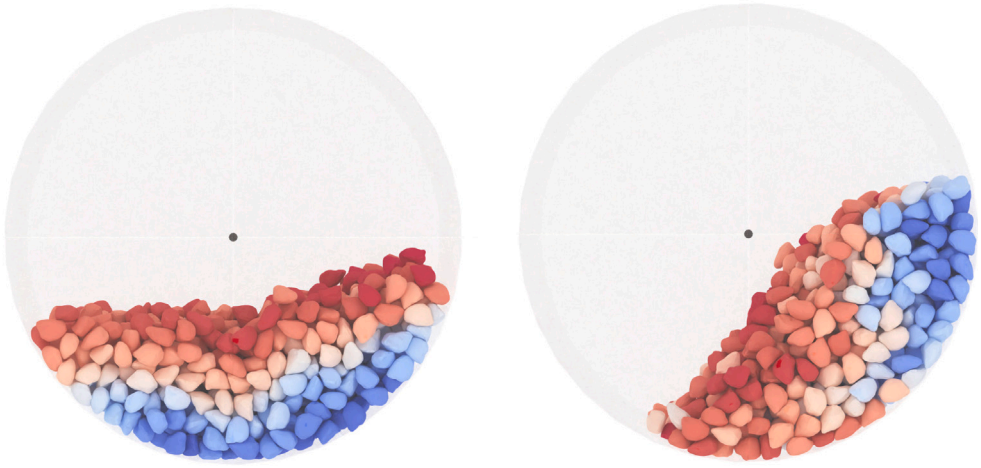


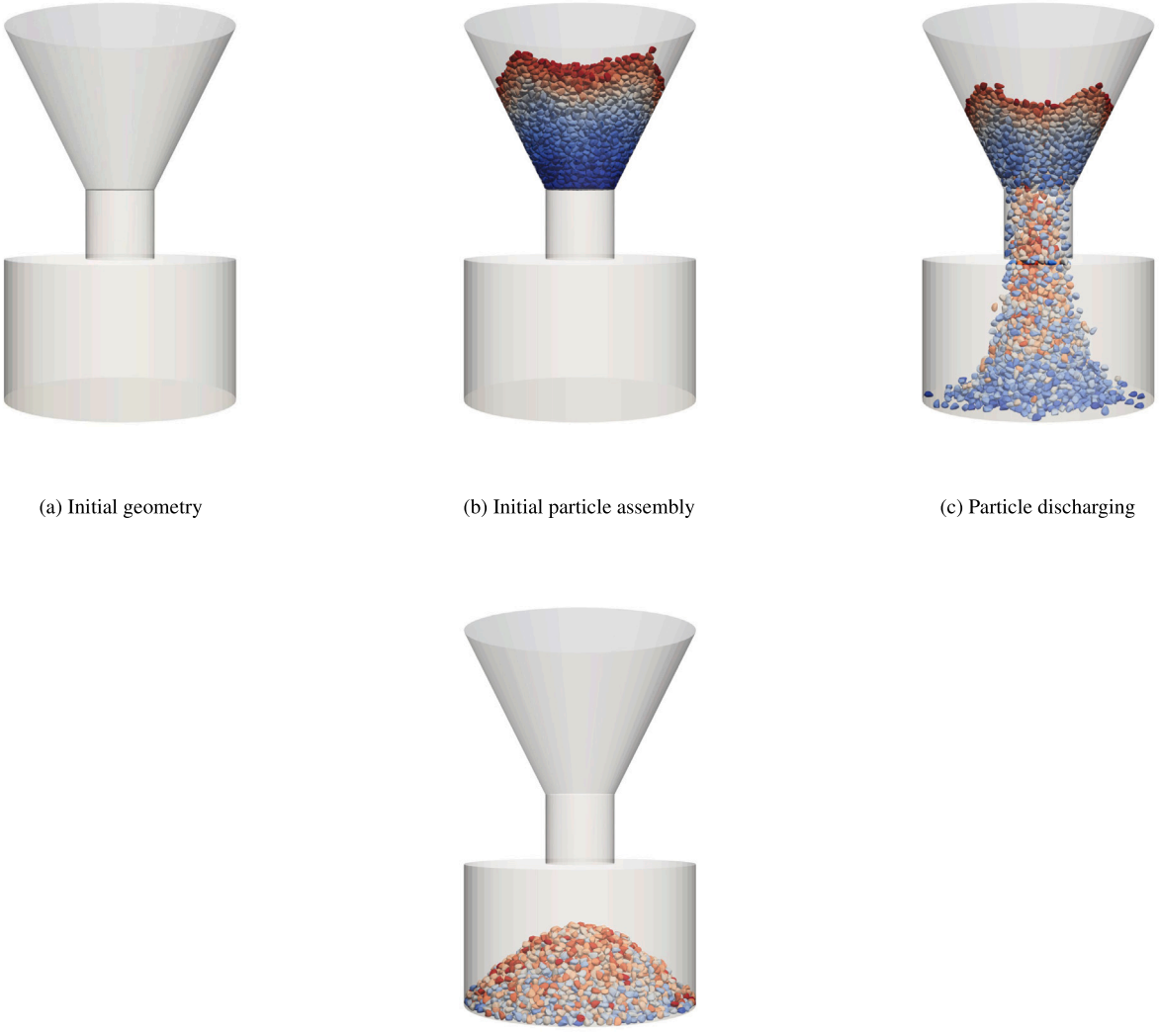
Fig. 25. Representative random packing snapshots from SDF-based DEM simulations with different particle shapes. The mix case includes particles of all four types.

of geometries. Compared with mesh- and voxel-based methods, NetSDF reduces per-particle memory usage by over 80%, and achieves comparable contact query performance with smoother numerical behavior. DEM simulations involving random packing, drum rotation, funnel discharge, and slope rockfall demonstrate robust contact detection and realistic macroscopic response. Parametric studies further reveal that moderate network architectures (5–6 hidden layers, 30–50 neurons per layer) and latent dimensionality (8–12) are sufficient for reliable encoding. Irregular shapes exhibit higher sensitivity to training density, but remain tractable under the proposed formulation.



(a) After packing (b) During rotation

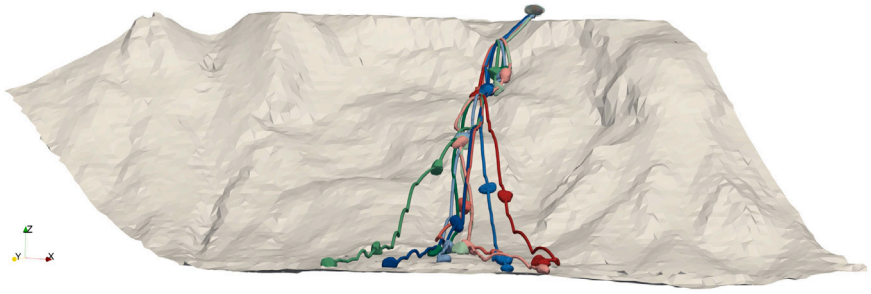
Fig. 26. Snapshots of rotating drum simulation with CodedNetSDF particles.



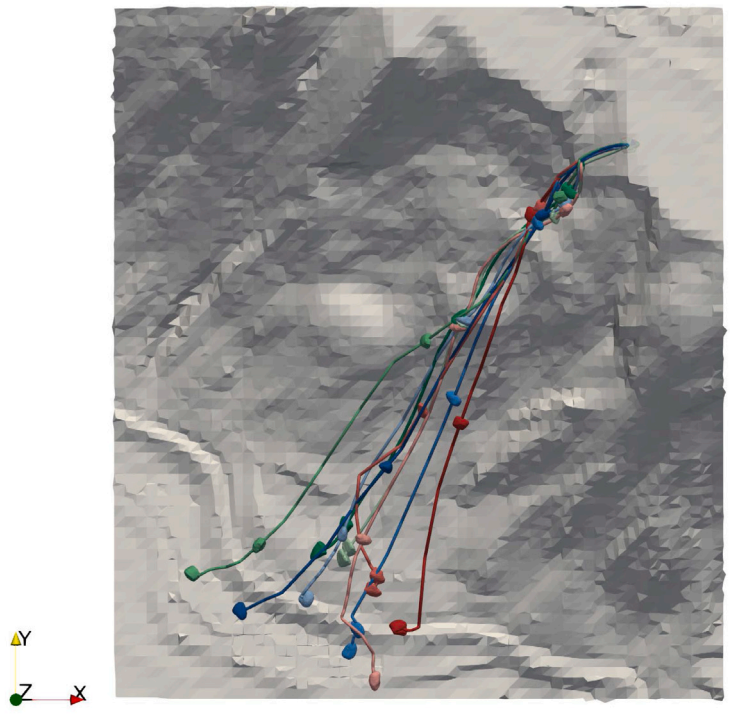
(a) Initial geometry (b) Initial particle assembly (c) Particle discharging

(d) Final particle assembly

Fig. 27. Snapshots of funnel discharge simulation.



(a) Perspective view of the terrain mesh.



(b) Top-down view of lateral dispersion.

Fig. 28. Comparison of simulated rockfall trajectories on a realistic slope for different particle generation strategies.

As a learnable and differentiable representation, NetSDF offers unique advantages beyond conventional geometric models. Its continuous structure supports not only shape reconstruction and simulation, but also virtual shape generation and latent space interpolation. The network parameters and latent codes are amenable to gradient-based optimization, making the framework well suited for topology-aware shape design, inverse modeling, and integration with differentiable solvers. These capabilities position NetSDF as a compact, extensible, and simulation-compatible foundation for data-driven particle mechanics, with potential applications in material design, microstructure-informed modeling, and generative computational mechanics.

CRediT authorship contribution statement

Chenghao Li: Writing – original draft, Validation, Formal analysis, Data curation. **Zhengshou Lai:** Writing – review & editing, Software, Conceptualization. **Shuai Huang:** Validation, Software, Formal analysis, Data curation. **Linchong Huang:** Supervision, Resources, Project administration.

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.

Acknowledgments

This work was supported by the Guangdong Major Project of Basic Research (2025B0303000011), the National Natural Science Foundation of China (52478428), and the Shenzhen Kunpeng Young Innovative and Entrepreneurial Project (KPGR20250731183203033).

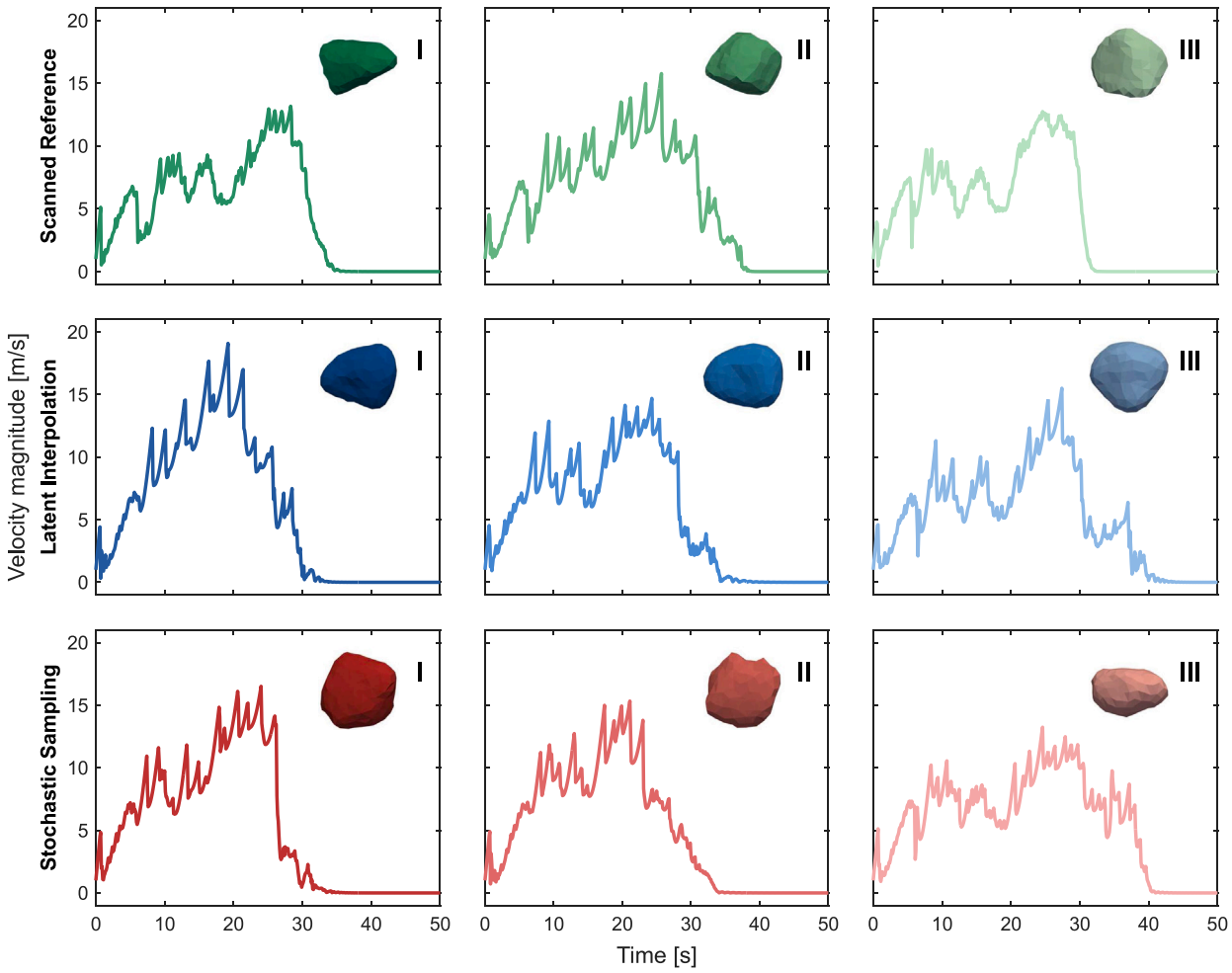


Fig. 29. Matrix of velocity magnitude time-histories for representative particle samples.

Data availability

Data will be made available on request.

References

Argilaga, A., Zhao, C., Li, H., Lei, L., 2024. Synthesising microstructures of a partially frozen salty sand using voxel-based 3D generative adversarial networks. Comput. Geotech. 170, 106247.

Atzmon, M., Lipman, Y., 2020a. Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2565–2574.

Atzmon, M., Lipman, Y., 2020b. Sald: Sign agnostic learning with derivatives. arXiv preprint arXiv:2006.05400.

Barr, A.H., 1981. Superquadrics and angle-preserving transformations. IEEE Comput. Graph. Appl. 1 (01), 11–23.

Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P., 2021. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5855–5864.

Blott, S.J., Pye, K., 2008. Particle shape: a review and new methods of characterization and classification. Sedimentology 55 (1), 31–63.

Carrasco, S., Cantor, D., Ovalle, C., Dubois, F., 2025. Particle shape distribution effects on the critical strength of granular materials. Comput. Geotech. 177, 106896.

Chen, J., Indraratna, B., Vinod, J.S., Ngo, T., Liu, Y., 2025. Effects of particle shape on the shear behavior and breakage of ballast: A DEM approach. Int. J. Geomech. 25 (1), 04024304.

Chen, Z., Zhang, H., 2019. Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948.

Chibane, J., Alldieck, T., Pons-Moll, G., 2020. Implicit functions in feature space for 3d shape reconstruction and completion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6970–6981.

Chibane, J., Pons-Moll, G., 2020. Implicit feature networks for texture completion from partial 3d data. In: European Conference on Computer Vision. Springer, pp. 717–725.

Cho, G.C., Dodds, J., Santamarina, C.J., 2006. Particle shape effects on packing density, stiffness, and strength: natural and crushed sands. J. Geotech. Geoenviron. Eng. 132, 591–602.

Cúñez, F.D., Patel, D., Glade, R.C., 2024. How particle shape affects granular segregation in industrial and geophysical flows. Proc. Natl. Acad. Sci. 121 (6), e2307061121.

Cundall, P.A., Strack, O.D., 1979. A discrete numerical model for granular assemblies. Géotechnique 29 (1), 47–65.

Curtin, R.R., Edel, M., Shrit, O., Agrawal, S., Basak, S., Balamuta, J.J., Birmingham, R., Dutt, K., Eddelbuettel, D., Garg, R., et al., 2023. mlpack 4: a fast, header-only C++machine learning library. arXiv preprint arXiv:2302.00820.

Duriez, J., Bonelli, S., 2021. Precision and computational costs of Level Set-Discrete Element Method (LS-DEM) with respect to DEM. Comput. Geotech. 134, 104033.

Feng, Y.T., 2021. An energy-conserving contact theory for discrete element modelling of arbitrarily shaped particles: Basic framework and general contact model. Comput. Methods Appl. Mech. Engrg. 373, 113454.

Feng, Y., 2023. Thirty years of developments in contact modelling of non-spherical particles in DEM: a selective review. Acta Mech. Sin. 39 (1), 722343.

Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y., 2020. Implicit geometric regularization for learning shapes. arXiv preprint arXiv:2002.10099.

van der Haven, D.L.H., Fragkopoulos, L.S., Elliott, J.A., 2023. A physically consistent discrete element method for arbitrary shapes using volume-interacting level sets. Comput. Methods Appl. Mech. Engrg. 414, 116165.

Huang, S., Huang, L., Lai, Z., Zhao, J., 2023. Morphology characterization and discrete element modeling of coral sand with intraparticle voids. Eng. Geol. 315, 107023.

Huang, S., Wang, P., Lai, Z., Yin, Z., Huang, L., Xu, C., 2024. Machine-learning-enabled discrete element method: The extension to three dimensions and computational issues. Comput. Methods Appl. Mech. Engrg. 432, 117445.

Jerves, A.X., Kawamoto, R.Y., Andrade, J.E., 2017. A geometry-based algorithm for cloning real grains. Granul. Matter 19 (2), 30.
Kawamoto, R., Andò, E., Viggiani, G., Andrade, J.E., 2016. Level set discrete element method for three-dimensional computations with triaxial case study. J. Mech. Phys. Solids 91, 1–13.
Kawamoto, R., Andò, E., Viggiani, G., Andrade, J.E., 2018. All you need is shape: Predicting shear banding in sand with LS-DEM. J. Mech. Phys. Solids 111, 375–392.
Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., 2023. 3D gaussian splatting for real-time radiance field rendering.. ACM Trans. Graph. 42 (4), 1–14, Article No. 139.
Lai, Z., Chen, Q., Huang, L., 2022a. Machine-learning-enabled discrete element method: Contact detection and resolution of irregular-shaped particles. Int. J. Numer. Anal. Methods Geomech. 46 (1), 113–140.
Lai, Z., Feng, Y., Zhao, J., Huang, L., 2024. Unifying the contact in signed distance field-based and conventional discrete element methods. Comput. Geotech. 176, 106764.
Lai, Z., Huang, L., 2021. A polybézier-based particle model for the DEM modeling of granular media. Comput. Geotech. 134, 104052.
Lai, Z., Zhao, S., Zhao, J., Huang, L., 2022b. Signed distance field framework for unified DEM modeling of granular media with arbitrary particle shapes. Comput. Mech. 70 (4), 763–783.
Lai, Z., Zhao, J., Zhao, S., Huang, L., 2023. Signed distance field enhanced fully resolved CFD-DEM for simulation of granular flows involving multiphase fluids and irregularly shaped particles. Comput. Methods Appl. Mech. Engrg. 414, 116195.
Li, M., Duan, Y., Zhou, J., Lu, J., 2023. Diffusion-SDF: Text-to-shape via voxelized diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12642–12651.
Maaten, L.v.d., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (Nov), 2579–2605.
Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A., 2019. Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4460–4470.
Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. Commun. ACM 65 (1), 99–106.
Mollon, G., Zhao, J., 2013. Generating realistic 3D sand particles using Fourier descriptors. Granul. Matter 15 (1), 95–108.
Moncada, R., Gupta, M., Thompson, A., Andrade, J.E., 2023. Level set discrete element method for modeling sea ice floes. Comput. Methods Appl. Mech. Engrg. 406, 115891.
Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A., 2020. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3504–3515.
Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S., 2019. DeepSDF: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174.
Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., Geiger, A., 2020. Convolutional occupancy networks. In: European Conference on Computer Vision. Springer, pp. 523–540.
Project, T.C., 2024. CGAL User and Reference Manual, 6.0.1 CGAL Editorial Board, URL https://doc.cgal.org/6.0.1/Manual/packages.html.
Qiao, T., Wang, S., Ji, S., 2025. A bonded polyhedral DEM model for irregular cemented granular materials based on energy-conserving contact theory. Granul. Matter 27 (3), 81.
Qin, S.-W., Liu, X.-W., Peng, S.-Y., Zhang, C.-B., Yao, J.-Y., Lv, J.-F., Wan, F., 2025. Effect of rockfall shape on the coefficient of restitution: Insights from laboratory rockfall multi-parameter experiments. Rock Mech. Rock Eng. 1–25.
Radvilaitė, U., Ramírez-Gómez, Á., Kačianauskas, R., 2016. Determining the shape of agricultural materials using spherical harmonics. Comput. Electron. Agric. 128, 160–171.
Shi, J., Zhang, W., Wang, W., Sun, Y., Xu, C., Zhu, H., Sun, Z., 2021. Randomly generating three-dimensional realistic schistous sand particles using deep learning: Variational autoencoder implementation. Eng. Geol. 291, 106235.
Sitzmann, V., Chan, E., Tucker, R., Snavely, N., Wetzstein, G., 2020. Metasdf: Meta-learning signed distance functions. Adv. Neural Inf. Process. Syst. 33, 10136–10147.
Tan, P., Sitar, N., 2024. Parallel implementation of LS-DEM with hybrid MPI+ OpenMP. Comput. Geotech. 172, 106408.
Wang, S., Liu, L., Xu, Q., Liang, D., Ji, S., 2024. A unified Minkowski sum model for largely deformed granular materials with arbitrary morphologies. Comput. Methods Appl. Mech. Engrg. 432, 117427.
Wang, X., Yin, Z.Y., Xiong, H., Su, D., Feng, Y.T., 2021. A spherical-harmonic-based approach to discrete element modeling of 3D irregular particles. Internat. J. Numer. Methods Engrg. 122 (20), 5626–5655.
Wu, T., Yuan, Y.-J., Zhang, L.-X., Yang, J., Cao, Y.-P., Yan, L.-Q., Gao, L., 2024. Recent advances in 3d gaussian splatting. Comput. Vis. Media 10 (4), 613–642.
Xi, J., Gao, L., Zheng, J., Wang, D., Wang, G., Guan, Z., Zheng, J., 2025. Gravel particle shape classification from half-particle point clouds using a dynamic graph edge convolution neural network. Comput. Geotech. 179, 107015.
Xiong, W., Wang, J., 2021. Gene mutation of particle morphology through spherical harmonic-based principal component analysis. Powder Technol. 386, 176–192.
Xiong, W., Wang, J., Wu, M., 2022. Effects of morphological gene decay and mutation on the micro–macro mechanical behaviours of granular soils. Géotechnique 74 (11), 1076–1094.
Xiong, H., Zhang, Z., Bao, X., Wu, H., Yin, Z.-y., Chen, X., 2024. Micro-mechanical analysis of particle shape effect on suffusion of gap-graded soils. Comput. Geotech. 165, 105925.
Xu, J., Li, Z., Du, B., Zhang, M., Liu, J., 2020. Reluplex made more practical: Leaky relu.
Yin, Z., Wang, P., Dai, S., 2023. Microstructures and micromechanics of geomaterials. J. Zhejiang Univ.-Sci. A 24 (4), 299–302.
Yoshida, T., Nomura, R., Tuda, Y., Yoshida, I., Terada, K., Moriguchi, S., 2024. Modeling effect of rock shape characteristics on run-out distribution of rockfalls. Soils Found. 64 (6), 101531.
Zhan, L., Peng, C., Zhang, B., Wu, W., 2021. A surface mesh represented discrete element method (SMR-DEM) for particles of arbitrary shape. Powder Technol. 377, 760–779.
Zhang, P., Dong, Y., Galindo-Torres, S.A., Scheuermann, A., Li, L., 2021. Metaball based discrete element method for general shaped particles with round features. Comput. Mech. 67, 1243–1254.
Zhang, J., Willis, A.R., 2024. Bridging formal shape models and deep learning: A novel fusion for understanding 3D objects. Sensors 24 (12), 3874.
Zhao, Y., Gao, X., Zhang, P., Lei, L., Li, S.Z., Galindo-Torres, S., 2023a. Reconstruction and generation of 3D realistic soil particles with metaball descriptor. Comput. Geotech. 161, 105564.
Zhao, S., Zhao, J., 2019. A poly-superellipsoid-based approach on particle morphology for DEM modeling of granular media. Int. J. Numer. Anal. Methods Geomech. 43 (13), 2147–2169.
Zhao, J., Zhao, S., Luding, S., 2023b. The role of particle shape in computational modelling of granular matter. Nat. Rev. Phys. 5 (9), 505–525.
Zheng, Z., Yu, T., Dai, Q., Liu, Y., 2021. Deep implicit templates for 3d shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1429–1439.
Zhou, B., Wang, J., Zhao, B., 2015. Micromorphology characterization and reconstruction of sand particles using micro X-ray tomography and spherical harmonics. Eng. Geol. 184, 126–137.
Zhu, Z., Qu, T., Zhao, J., 2025. Single-Image 3D Particle Reconstruction Via Generative AI-Empowered Large Vision Models, vol. 40, (28), pp. 5288–5306.