Data-driven model for soil-object interaction with large plastic deformation

Qinghao Yang[a], James P. Hambleton[b]

[a]*Department of Civil and Environmental Engineering, Northwestern University, Evanston, IL, USA*
[b]*Department of Engineering, University of Cambridge, Cambridge, UK*

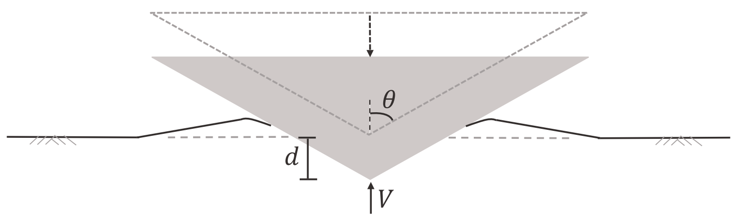
Abstract

Soil undergoes large deformation in many soil-object interactions, such as indentation and ploughing. Modeling these processes presents challenges due to the large and predominantly plastic deformation and the associated free surface evolution. This paper proposes a data-driven method to model these processes efficiently. In this method, neural network models are employed to predict force-displacement relationships throughout the deformation process, accounting for the evolution of the free surface. Proper orthogonal decomposition (POD) facilitates the low-dimensional representation of surface geometries. The accuracy and efficiency of the method are demonstrated by applying the approach to three examples: wedge indentation, cylinder indentation, and plate ploughing. By considering various object geometries, soil parameters, and motions, the study also showcases the method's potential for application across a wide range of problems characterized by soil-object interaction.

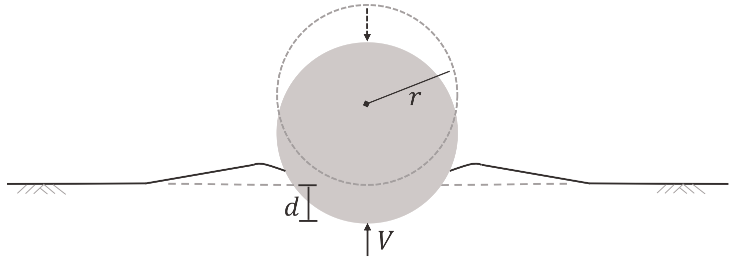
Keywords: soil-object interaction, large deformation, data-driven, neural network, proper orthogonal decomposition

1. Introduction

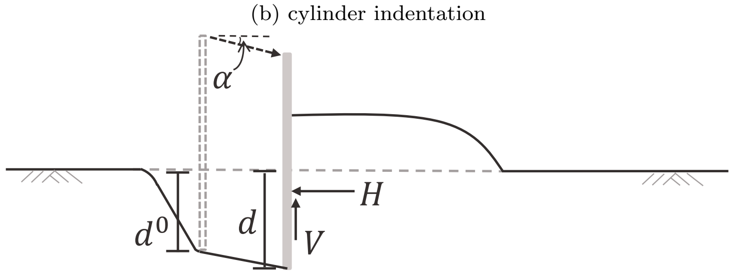
Soil-structure and soil-tool interaction are ubiquitous in geotechnical engineering. In many applications, including pile installation (Hamann et al., 2015; Dijkstra et al., 2011), lateral buckling of pipelines (Chatterjee et al., 2012; Wong et al., 2010), soil ploughing and cutting (McKyes and Ali, 1977; Yong and Hanna, 1977; Hambleton et al., 2014), and mobility of off-road vehicle (Hambleton and Drescher, 2008, 2009), the movement of structures or tools induces large plastic deformations, along with the evolution of free surfaces (Fig. 1). In turn, the changing free surfaces affect the resistive forces. Thus, it is crucial to account for large soil deformations when analyzing and modeling soil-structure/tool interaction. When it comes to the design of structures and tools interacting with soils, the key objective is to model the force-displacement history for the tool or structure throughout the whole process of large plastic deformation. Additionally, the models must be computationally efficient, as they are usually used repeatedly, particularly within control and optimization algorithms.



(a) wedge indentation



(b) cylinder indentation



(c) plate ploughing

Figure 1: Diagrams of soil-object interaction. The grey dashed lines represent the original free surfaces and original locations of objects. Variable d is the depth of objects. H and V are horizontal and vertical resistive forces, respectively. Angle θ represents half of the wedge's tip angle. Variable r represents the radius of the cylinder. Angle α represents the inclination angle of the plate velocity.

Theoretically, modeling the whole force-displacement history is challenging due to the complexities introduced by large plastic deformations, evolving free surfaces, and changing contact conditions. In geotechnical engineering, several analytical models were developed to study the mechanical behaviors of geo-structures such as foundations and anchors (Nova and Montrasio, 1991; Houlsby and Cassidy, 2002; Bienen et al., 2006; Chatzigogos et al., 2009). Nevertheless, those analytical models typically account only for the nonlinearity from material properties but do not consider the geometric nonlinearity caused by large deformation, while it significantly affects the force-displacement histories (Tian and Cassidy, 2011; Hambleton and Drescher, 2012).

Given the limitations of analytical models in handling large deformations, numerical methods are the predominant techniques for modeling such processes. The methods include Arbitrary-Lagrangian-Eulerian-based finite element method (Wang et al., 2015; Qiu et al., 2011), discrete element method (Tsuji et al., 2012; Shmulevich et al., 2007), material point method (Soga et al., 2015), etc. Although these methods can address the large plastic deformation and the corresponding evolution of free surfaces, they are computationally expensive for engineering practice. In design optimization, for example, iterative numerical simulations are necessary to determine the optimal design parameters. A more efficient alternative is to develop surrogate models that directly capture variables of interest—specifically, force-displacement relationships as discussed in this paper—using data-driven methods such as machine learning. Through machine learning, a surrogate model can be constructed after learning the statistical properties of a set of problems and then applied to predict similar problems.

The popularity of data-driven methods is supported by the surge of accessible data and the advancement of computing power over the past decade. Machine learning and deep learning have shown significant potential across various fields. In the physical and engineering domains, for example, Raissi et al. (2019) introduced physics-informed neural networks, which integrated physical governing equations with neural networks to solve partial differential equations from empirical data. In the pursuit of efficient modeling, Hesthaven and Ubbiali (2018) applied neural networks to reduce the computational time in high-fidelity models. Research on solid materials dates back to the 1990s when Ghaboussi et al. utilized feedforward neural networks to model the mechanical behaviors of concrete (Ghaboussi et al., 1991) and geomaterial (Ghaboussi et al., 1998; Hashash et al., 2004) based on experimental data. More recently, machine learning techniques were applied to model complex heterogeneous materials (Le et al., 2015; Bessa et al., 2017; Liu et al., 2019; Mozaffar et al., 2019). In the context of soil-structure interaction, Zhang et al. (2020) employed recurrent neural networks to learn force-displacement relations and predict the failure envelope of foundations. Despite these advances, there has been limited research on the effects of large deformations in soil-structure/tool interaction.

This paper proposes a data-driven approach for soil-structure/tool interaction involving large deformation. In this approach, neural network models are developed to capture the whole history of force-displacement relations. To effectively handle the evolution of free surfaces caused by large deformation, the geometries of free surfaces are taken into consideration when building the neural network models. Nevertheless, without any processing, the number of variables to represent a free surface is high. This creates a high-dimensional challenge that requires substantial amounts of data for training models. Obtaining such data from large-deformation numerical methods or experiments for soil-structure/tool interaction can be costly. Therefore, this paper reduces the dimensions representing a free surface before building the neural network models. The proposed models are evaluated for three fundamental problems involving large deformation: wedge indentation, cylinder indentation, and plate ploughing (Fig. 1).

2. Geometric data-driven method

2.1. Problem statement

In numerical methods, a standard approach to handle large deformations involves dividing them into a sequence of small deformations. For instance, in large deformation finite element analysis, small-strain finite element analysis is performed iteratively to update stress fields and geometries of the computational domain. To further address the mesh distortion caused by large deformation, the Arbitrary Lagrangian-Eulerian framework is often employed to replace the Lagrangian framework (Wang et al., 2015). In geotechnical engineering, several other techniques are also used to address large soil deformations and the associated changes in surface geometries. Sequential limit analysis, for example, applies finite element limit analysis to compute velocity fields and update surface geometries incrementally (Kong et al., 2017). Similarly, the sequential kinematic method applied the kinematic method in limit analysis to calculate velocity fields at each small deformation step (Shi et al., 2021). Regardless of the specific numerical method used, the most computationally intensive task is computing displacement or velocity fields at each small deformation step. To overcome this challenge, we applied a data-driven method to bypass these computational steps.

In our problems of soil-structure/tool interaction, structures or tools are treated as rigid objects. The quantities of interest are the displacements of a rigid object and the resistive forces (V and H in Fig. 1) on it. Finite element analysis is used to generate data for training and evaluating a data-driven model. Considering the convenience of performing displacement-control finite element analysis, the movement of a rigid body is displacement control. The predicted quantity is the resistive forces throughout a large deformation process. Therefore, the data-driven model must provide resistive forces as outputs for each small deformation step. The mechanical response in those steps is determined by soil properties, object displacements, object shapes, contact conditions, and surface geometries. All of those quantities can serve as the inputs to the data-driven model. Furthermore, as surface geometries keep changing throughout a large deformation process, the surface geometries must also be outputted by the data-driven model to serve as inputs for the subsequent small deformation step.

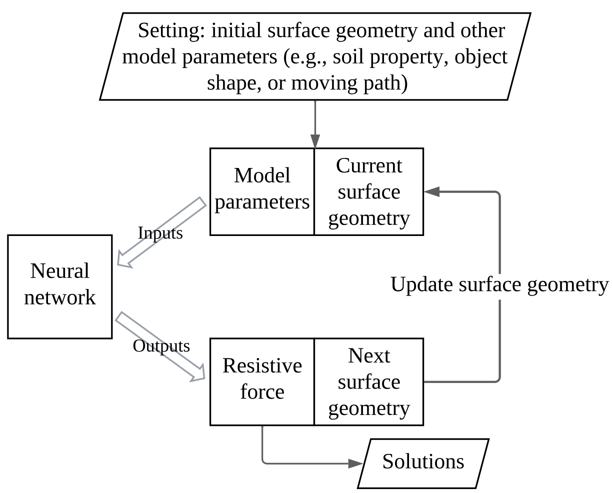


Figure 2: Prediction of resistive forces with the geometric data-driven method.

In summary, the inputs to the data-driven model include surface geometries and other parameters of interest, and the outputs are surface geometries and resistive forces (Fig. 2). It is crucial to use an appropriate mathematical expression for surface geometries, which are both the inputs and outputs of the data-driven model.

2.2. Low-dimensional approximation of free surfaces

This section describes the method for incorporating surface geometries into the data-driven model. A free surface is a continuous surface in which an infinite number of points can move freely, i.e., the degrees of freedom (DOFs) of the free surface are infinite. In numerical methods like finite element methods, the DOFs are reduced to a finite number through discretization. Despite lowering the DOFs of free surfaces to a finite number, the required sample size is still too high, making data generation through experiments or simulations impractical. This is due to the exponential increase in sampling points as the input dimension grows, a phenomenon known as the "curse of dimensionality". A simple math calculation is given to explain this phenomenon as follows.

In this work, we only focus on 2D problems, where the DOFs of a free surface are twice the number of discrete points. Assuming that free surfaces are expressed using 100 points, the inputs corresponding to free surfaces in a neural network will contain 200 variables. Each variable should be sampled sufficiently to ensure that the neural network is predictive. If 10 possible values are sampled for each variable, the number of samples would be 100^{10} to obtain dense data (sparse data makes it hard to learn relations among high-dimensional data). Even if generating one sample takes only several minutes through a small-strain finite element analysis, the total time to generate all those samples is unbearable. Therefore, before applying the data-driven method, it is essential to use lower DOFs to express the evolving free surfaces.

2.2.1. Proper orthogonal decomposition

The key to reducing the DOFs is to find a low-dimensional representation of the high-dimensional data while preserving its essential properties. Intuitively, a free surface can be approximated by piecewise linear functions, namely, several line segments. This approach does not account for the features of data. Proper orthogonal decomposition (POD) (Sirovich, 1987), also known as principle component analysis (PCA) in statistics, is a dimensionality reduction method based on features of data. Specifically, POD works to find common features of a group of high-dimensional data and then express the group of data in a low-dimensional space. POD has been widely applied in image and signal processing (Rosenfeld, 1976; Castells et al., 2007; Abdi and Williams, 2010) and reduced-order modeling (Rowley et al., 2004; Rozza et al., 2008; Benner et al., 2015).

The detailed implementation of POD is as follows. After collecting M snapshots (results from large deformation finite element analysis in this paper), a matrix $\boldsymbol{Y}^h \in \mathbb{R}^{N\times M}$ is constructed, i.e.,

$$\boldsymbol{Y}^h = \left[\boldsymbol{y}_1^h \mid \boldsymbol{y}_2^h \mid \cdots \mid \boldsymbol{y}_j^h \mid \cdots \mid \boldsymbol{y}_M^h\right]. \tag{1}$$

The vector $\boldsymbol{y}_j^h$ is the dimensionless vertical coordinates of the surface points with fixed horizontal coordinates. The matrix $\boldsymbol{Y}^h$ could be decomposed by singular value decomposition (SVD) as

$$\boldsymbol{Y}^h = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{W}^T. \tag{2}$$

The columns of $\boldsymbol{U} \in \mathbb{R}^{N\times N}$ and $\boldsymbol{W} \in \mathbb{R}^{M\times M}$ are the left and right singular vectors of $\boldsymbol{Y}^h$, respectively. $\boldsymbol{\Sigma} = \mathrm{diag}\left(\sigma_1, \sigma_2, \ldots, \sigma_M\right)$ is a rectangular diagonal matrix, where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_M \geq 0$ are the singular values of $\boldsymbol{Y}$. Taking the $r(<< M)$ left singular vectors that correspond to the r largest singular value, a truncated matrix $\boldsymbol{V}(= \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_r) \in \mathbb{R}^{N\times r}$ could be obtained. $\boldsymbol{V}$ are also called the reduced basis, constructing a reduced space. Let $\boldsymbol{Y}^l = \boldsymbol{V}^T\boldsymbol{Y}^h \in \mathbb{R}^{r\times M}$, then each column of $\boldsymbol{Y}^l$ is the projection of the original $\boldsymbol{y}_j^h$ onto the reduced space. The approximated solutions $\hat{\boldsymbol{Y}}^h$ in the original high-dimensional space can be obtained from the projections in low-dimensional space through

$$\hat{\boldsymbol{Y}}^h = \boldsymbol{V}\boldsymbol{Y}^l. \tag{3}$$

The least squares error of POD is expressed as

$$\left\|\boldsymbol{Y}^h - \hat{\boldsymbol{Y}}^h\right\|_F^2 = \sum_{j=1}^{M} \left\|\boldsymbol{y}_j^h - \boldsymbol{V}\boldsymbol{V}^T\boldsymbol{y}_j^h\right\|_2^2. \tag{4}$$

According to Eckart–Young–Mirsky theorem (Eckart and Young, 1936), the reduced basis $\boldsymbol{V}$ from SVD minimizes the least squares error among all the orthogonal basis $\boldsymbol{B} \in \mathbb{R}^{N\times r}$. i.e.,

$$\min \sum_{j=1}^{M} \left\|\boldsymbol{y}_j^h - \boldsymbol{B}\boldsymbol{B}^T\boldsymbol{y}_j^h\right\|_2^2 = \sum_{j=1}^{M} \left\|\boldsymbol{y}_j^h - \boldsymbol{V}\boldsymbol{V}^T\boldsymbol{y}_j^h\right\|_2^2 = \sum_{j=r+1}^{M} \sigma_j^2. \tag{5}$$

Because the error of POD is given by the sum of the squares of the singular values not included in the POD basis, the singular values provide guidance for choosing the size of the POD basis. A typical approach is to choose r so that

$$\frac{\sum_{j=1}^{r} \sigma_j^2}{\sum_{j=1}^{M} \sigma_j^2} > k \tag{6}$$

where k is a user-specified tolerance.

After selecting the reduced basis $\boldsymbol{V}(= \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_r)$, a vector $\boldsymbol{y}^h$ can be approximated as

$$\boldsymbol{y}^h \approx \boldsymbol{V}\boldsymbol{y}^l = \sum_{i=1}^{r} y_i^l \boldsymbol{v}_i. \tag{7}$$

During the training of data-driven models, we only need to use y_i^l, the coefficients of reduced basis $\boldsymbol{v}_i$, to represent surface geometries as both inputs and outputs of the models. In the following part of this paper, the coefficients are expressed using λ.

2.2.2. Unified expression of geometric data

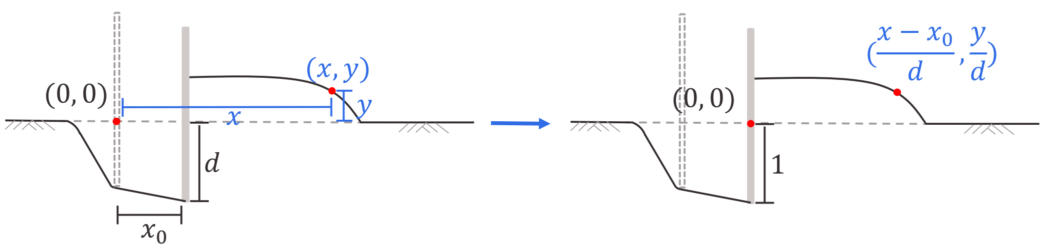


Figure 3: Dimensionless formulation for plate phoughing.

For convenience in model training, we expressed the geometric data in a unified form before performing POD. Concretely, the surface geometries are expressed in a dimensionless form, and the coordinates are relative to the rigid object. Taking the ploughing process of a plate as an example (Fig. 3), the plate is set as a reference. The dimensionless coordinates of points on the surface and the corresponding dimensionless force are set as

$$\hat{x} = \frac{x - x_0}{d}, \quad \hat{y} = \frac{y}{d}, \quad \hat{H} = \frac{H}{cd}, \tag{8}$$

where (x, y) are the coordinates of surface points in the original system, x_0 is the horizontal coordinate, d is the depth of the plate, c is the cohesion of soil, and H is the horizontal resistive force on the plate. $\hat{x}, \hat{y}, \hat{H}$ are the dimensionless forms of x, y, H. To keep the number and horizontal coordinates of all surface nodes fixed, we interpolated the vertical coordinates $[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N]$ of surface points at the fixed horizontal coordinates $[\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_N]$ when collecting data. Here, N represents the number of vertical coordinates stored in the database to represent free surfaces.

2.3. Neural network

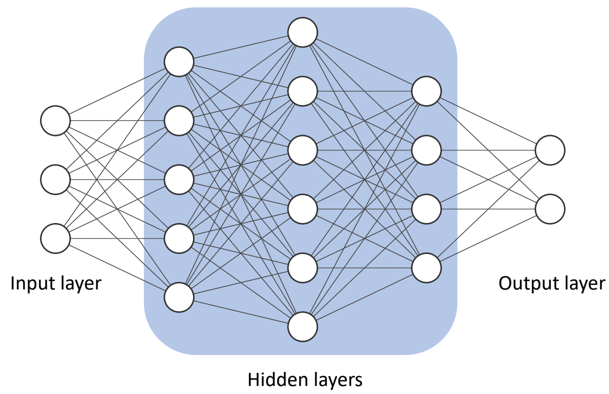


Figure 4: An example of a feedforward neural network with 3 input neurons, 3 hidden layers, and 2 output neurons.

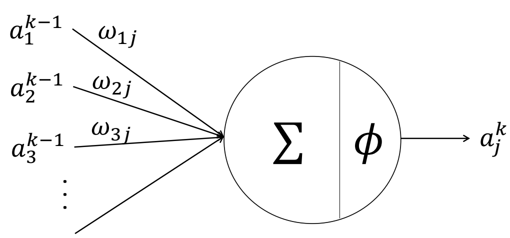


Figure 5: A neuron in feedforward neural network.

The data-driven model is built using a feedforward neural network (FNN), which is a fundamental type of neural network. In FNN, data flows from an input layer, through hidden layers, and to an output layer (Fig. 4). The neurons—the nodes in the network—are fully connected across adjacent layers. For a neuron j in the k layer, its output is determined by all outputs in the $k-1$ layer (Fig. 5) as

$$a_j^k = \phi\left(\sum_{i=1}^{N} w_{ij} a_i^{k-1} + b_i^{k-1}\right), \tag{9}$$

where N is the total number of neurons in the $k-1$ layer, a_i^{k-1} is the output from the neurons in that layer. w_{ij} is the weight of the connection (a_j^k, a_i^{k-1}), and b_i^{k-1} is the bias analogous to the constant in a linear function. w_{ij} and b_i^{k-1} express a linear transformation. ϕ is the activation function that decides to what extent the neuron j in the k layer is activated. Mathematically, activation functions enable neural networks to approximate nonlinear behaviors. Cybenko (1989) proved that 2-hidden-layer networks with differentiable activation functions can approximate any nonlinear functions. In this work, the ReLU activation function (Brownlee, 2019) is employed.

Training a neural network is an optimization problem aimed at finding the optimal weights minimizing a loss function, which measures the discrepancy between the neural network's outputs and the actual outputs for the same inputs. The training process involves iteratively updating weights until the loss function falls below a specified threshold. In this paper, the mean squared error (MSE) is used as the loss function, and the Adam optimizer (Kingma and Ba, 2014) is used for optimization.

2.4. Framework of geometric data-driven method

In closing, the framework of the geometric data-driven method includes three phases as follows.

1. **Data generation:** The data in this paper comes from numerical solutions of RITSS method (Hu and Randolph, 1998). RITSS resolves the issue of mesh distortion caused by large deformation through the arbitrary Lagrangian-Eulerian technique, i.e., the displacements of mesh nodes are not identical to material points. Specifically, the distorted elements after a small deformation step are remeshed, and state variables are mapped onto new mesh nodes. It is convenient to integrate RITSS into a small-strain FEM solver. In this work, Abaqus (Systèmes, 2016) is used for small-strain simulations, and Python scripts were developed to control the remeshing and mapping processes. This paper explores the data-driven approach for perfect plasticity, i.e., there is no hardening or softening during the process of plastic deformation.

2. **Model training:** Neural networks are trained to develop surrogate models for the force-displacement relations during large plastic deformation. After performing POD on free surfaces, the inputs of neural networks are the coefficients $\boldsymbol{\lambda}^t$ of reduced basis and other parameters $\boldsymbol{p}$ of interest (e.g., geometric parameters of objects, material parameters of soils, and displacement vectors). The outputs include resistive forces $\boldsymbol{F}^t$ and the updated coefficients $\boldsymbol{\lambda}^{t+1}$ for the next step. Before training the neural network, all input and output variables are scaled through standardization,
$$z = \frac{z^o - \bar{z}}{\sigma}, \tag{10}$$
where z^o is an original value of a variable, $\bar{z}$ is the mean of all values in training data, and σ is the standard deviation of all values. The corresponding loss function, expressed in mean squared errors (MSE), is given by
$$MSE = \frac{1}{(M_1 + M_2)N} \sum_{i=1}^{N} (\sum_{j=1}^{M_1} \left(f'_{i,j} - f_{i,j}\right)^2 + \sum_{j=1}^{M_2} \left(\lambda'_{i,j} - \lambda_{i,j}\right)^2) \tag{11}$$
where N is the number of data points, M_1 is the dimension of resistive forces, and M_2 is the number of coefficients, i.e., the dimension of reduced basis.
3. **Prediction:** After the model is trained, the resistive forces can be predicted given the corresponding parameters as shown in Fig. 2. In each small deformation step, the neural network model can generate a resistive force and a new set of coefficients of reduced basis. Taking the new set of coefficients as inputs of the neural network model, the resistive force for the next step is generated. Consequently, a sequence of resistive forces during the whole process of large deformation can be generated iteratively.

3. Results

To investigate the effectiveness of the proposed method, we investigated three examples (Fig. 1) involving large deformation. In the first example, we applied the data-driven method to predict the indentation of wedges with various geometries. The second example was cylinder indentation which tested the effectiveness of the data-driven method in predicting force-displacement relations under different soil properties. The third example focused on the ploughing processes of a plate. The plate ploughing tested the effectiveness of the data-driven method in predicting force-displacement relations under various motions.

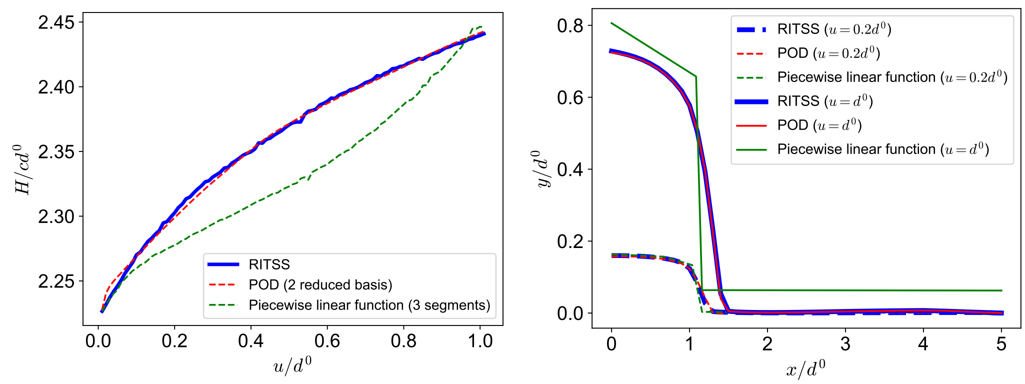
In the three examples, wedges, cylinders, and plates were all assumed to be rigid objects. The soil was ponderable with a density of 2000 kg/m^3. Young's modulus was 200 MPa. Poisson's ratio was taken as 0.3. Coulomb's law was used for the contact properties between soil and rigid objects and the coefficient of friction was 0.5. In the wedge indentation and plate ploughing, soils were cohesive materials modeled using the Tresca yield criterion with cohesion $c = 20$ kPa. In the cylinder indentation, soils were cohesive-frictional material modeled with the Mohr–Coulomb yield criterion. The flow rule was associative for all three examples. In the RITSS-based FEM method, finite element types were bi-linear rectangles. The size of elements was around 0.05 m during remeshing steps in RITSS. To avoid the effects of boundaries, the distance of boundaries to the rigid objects was taken large enough. For example, in wedge indentation tests, the distance of the bottom boundary to the wedge tip was 5 times the final depth of the wedge, and the distance of the lateral boundary to the wedge tip was $5 \tan\theta$ times the final depth, where θ is half of the tip angle.

3.1. Compared with piecewise linear approximation

Before testing the effectiveness of the neural network models, we compared the approximating effects of POD and piecewise linear functions. A horizontal ploughing process of a plate with depth d^0 was run through the RITSS method as a reference. To compare the approximating effects, after each small-strain analysis, the updated free surface was approximated with POD and piecewise linear functions, respectively. Then, the approximated surface was used as the geometric configuration of the next small-strain step. The error of force-displacement relations caused by this geometric approximation is shown in Fig. 6. Figure 6a shows that POD gives better approximations with lower dimensions compared to piecewise linear functions. Figure 6b shows that the error in surface geometries increases as the plate advances when piecewise linear functions are used.

3.2. Wedge indentation

In wedge indentation tests, various geometries of intruders, i.e., the wedge tip angles, were used as input parameters for the neural network model (Fig.



(a) force (H)-displacement (u) relations

(b) surface geometries after motion $0.2d^0$ and $1d^0$

Figure 6: Comparison between POD and piecewise linear approximation, where c represents cohesion and d^0 represents the depth of the plate.

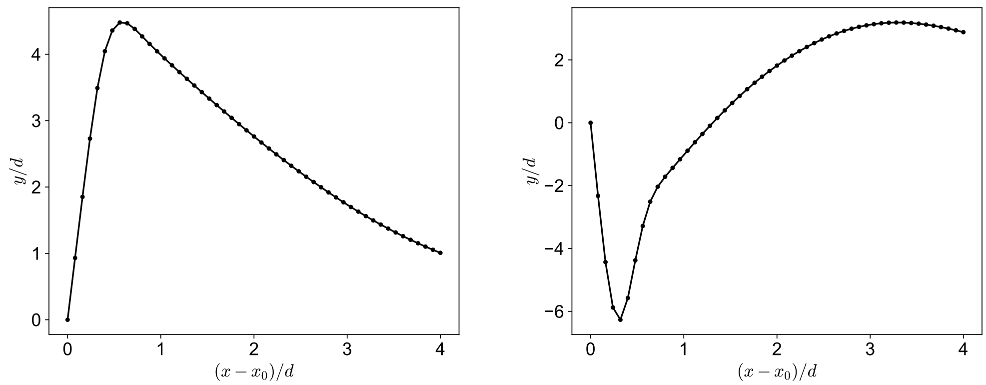
1a).

Finite element solutions for different tip angles ($\tan\theta = 1.5, 2.0, 2.5, 3.0, 3.5$) were collected to train the neural network. The final indentation depth was taken as 1 m for all simulations. The simulation results were collected every 0.01 m indentation, resulting in 100 data points per simulation. The dimensionless coordinates of surface points were calculated with Eq. 8. Applying POD on the dimensionless form of surface geometries and setting the tolerance to 99%, 2 reduced basis vectors were obtained (Fig. 7). As a result, the inputs of the neural network were the wedge tip angle θ and 2 coefficients of reduced basis λ_1, λ_2.

After training the neural network, tests were conducted for 4 new wedge tip angles ($\tan\theta = 1.6, 1.8, 2.3, 2.8$). Figure 8a compares the vertical forces on wedges predicted by the neural network with those computed via FEM. Figure 8b shows the relative errors of the predicted forces, calculated as

$$\epsilon = \left| \frac{F^{POD-NN} - F^{FEM}}{F^{POD-NN}} \right|, \tag{12}$$

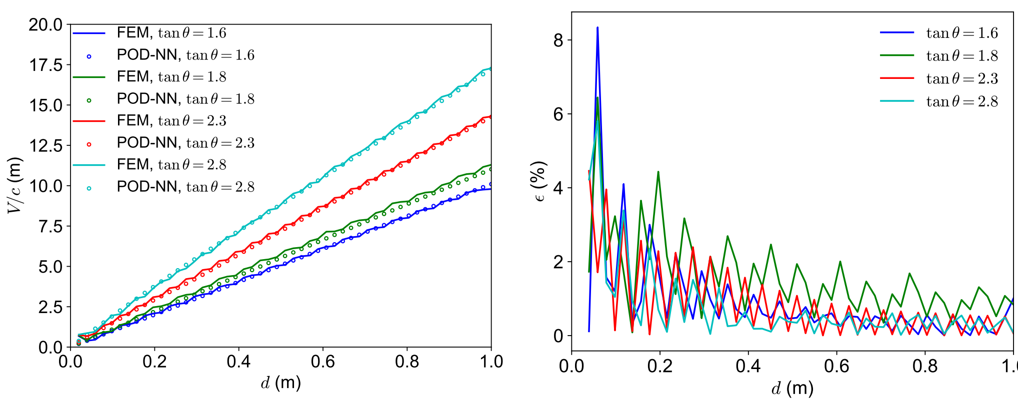
where F^{POD-NN} and F^{FEM} are the predicted force and the numerically solved force, respectively. In this example, the errors are calculated for vertical forces (V). The results show that the data-driven model gives a good prediction for force-displacement relations during wedge indentation. The relatively large errors observed at the initial stages are due to the incomplete contact between soil and wedges, causing inaccuracies in the simulations.



(a) The 1st reduced basis vector $\boldsymbol{v}_1$

(b) The 2nd reduced basis vector $\boldsymbol{v}_2$

Figure 7: Reduced basis for surface geometries during wedge indentation.

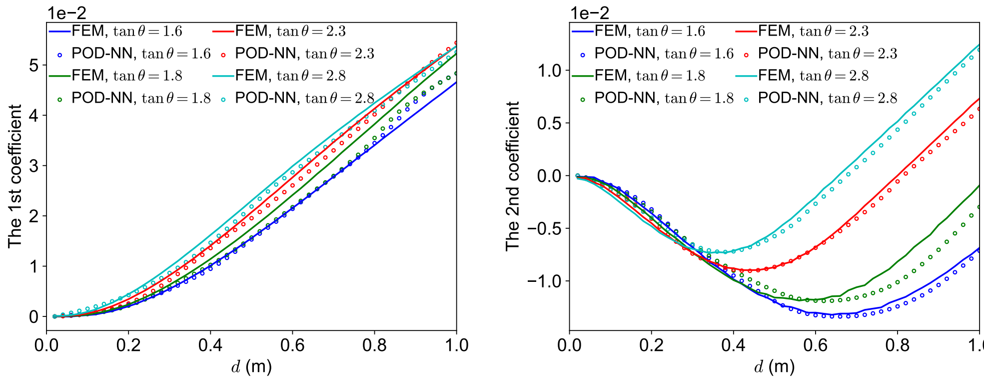


(a) Comparison of forces

(b) Relative errors of forces

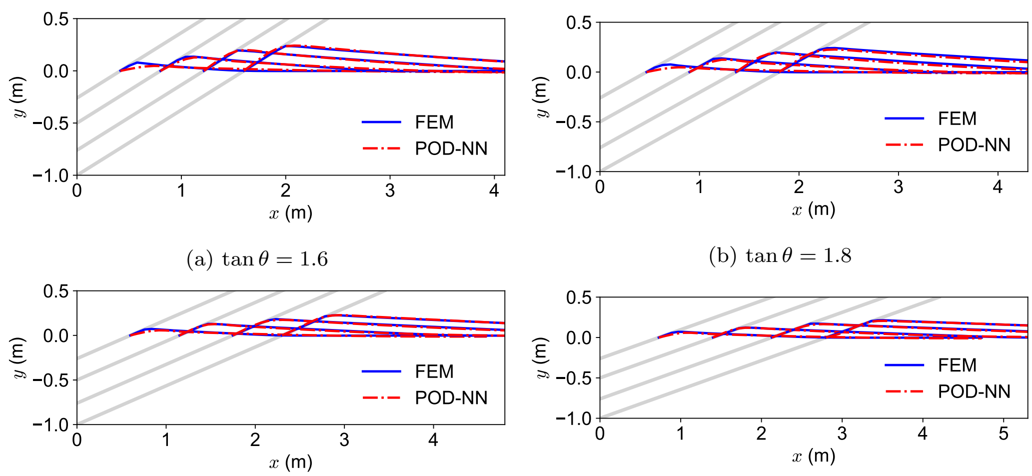
Figure 8: Vertical resistive force versus vertical displacement during wedge indentation.

Figure 9 compares the 2 coefficients of reduced basis. The 1st coefficient λ_1 has a larger magnitude than the 2nd one λ_2. This indicates that the 1st reduced basis vector $\boldsymbol{v}_1$ captures the dominant features of surface geometries, while the 2nd reduced basis vector $\boldsymbol{v}_2$ provides refinements. As stated in the methods section, high-dimensional surface geometries can be reconstructed from the predicted coefficients with $\lambda_1\boldsymbol{v}_1 + \lambda_2\boldsymbol{v}_2$. Figure 10 compares the reconstructed free surfaces with those computed via FEM, which illustrates that the evolution of free surfaces can be closely predicted by the data-driven model.



(a) (b)

Figure 9: Comparison of the coefficients of reduced basis during wedge indentation.



(a) $\tan\theta = 1.6$ (b) $\tan\theta = 1.8$

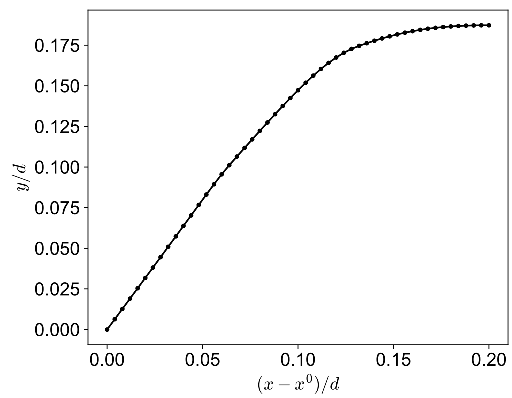
(c) $\tan\theta = 2.3$ (d) $\tan\theta = 2.8$

Figure 10: Comparison of free surfaces during wedge indentation. The surface geometries are shown when the indentation depths are 0.25 m, 0.5 m, 0.75 m, and 1 m. The gray lines are the locations of wedges, motion from top to bottom.

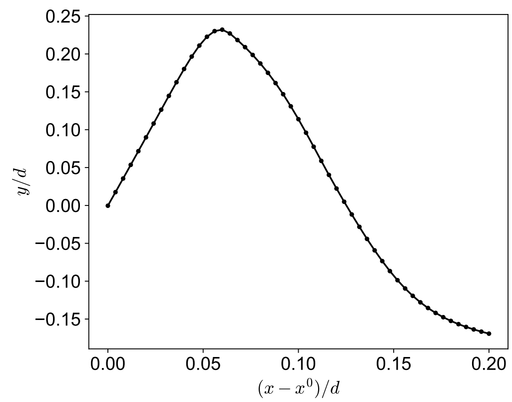
3.3. Cylinder indentation

In cylinder indentation (Fig. 1b), the additional variable besides coefficients of reduced basis was the friction angle (ϕ) of soil, whose cohesion was fixed as 20 kPa.

Finite element solutions for $\phi = 2°, 4°, 6°, 8°, 10°, 12°, 14°, 16°, 18°$ were collected to train neural networks. In each finite element simulation, the final vertical displacement was $0.2r$ (0.2 times the radius of the cylinder). 100 data points were collected from each simulation. Applying POD on the dimensionless form of surface geometries and setting the tolerance to 99.9%, 2 reduced basis vectors were generated (Fig. 11). Hence, the inputs of the neural network were the internal friction angle ϕ and 2 coefficients of reduced basis λ_1, λ_2.



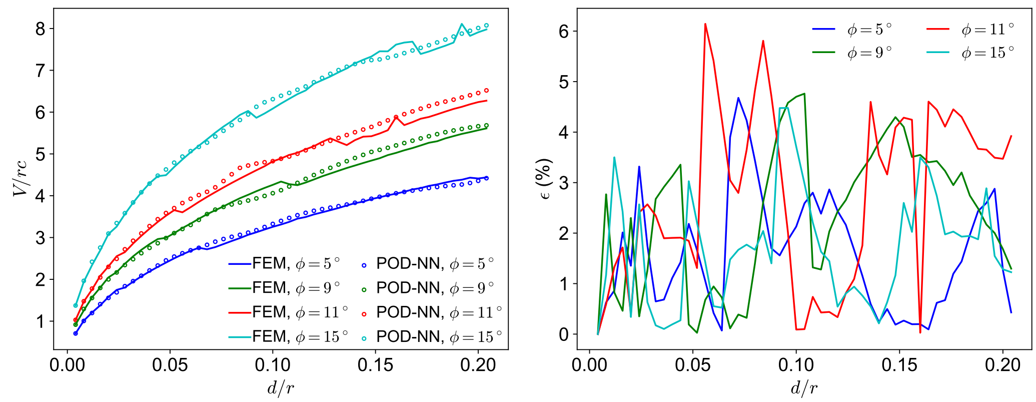
(a) The 1st reduced basis vector $\boldsymbol{v}_1$



(b) The 2nd reduced basis vector $\boldsymbol{v}_2$

Figure 11: Reduced basis for surface geometries during cylinder indentation.

When simulating cylinder indentations using the RITSS method, noticeable fluctuations appeared in the force-displacement curves. This was due to the sensitivity of the force response to the mesh, as the RITSS method involves sequential remeshing. To avoid overfitting the fluctuations, L1 penalty (Tibshirani, 1996) was added to the loss function when training the relationship between inputs and vertical forces. After training the neural network, tests were conducted for 4 new internal friction angles ($\phi = 5°, 9°, 11°, 15°$). Figure 12 compares the vertical forces on cylinders predicted by the neural network to those computed via FEM. The results demonstrate that the data-driven model accurately predicts the force-displacement relationships during cylinder indentation. Figure 13 compares the 2 coefficients of reduced basis. Figure 14 illustrates reconstructed free surfaces from the predicted coefficients alongside the FEM-computed surfaces, showing that the evolution of free surfaces can be captured effectively by the data-driven model.



(a) Comparison of forces

(b) Relative errors of forces

Figure 12: Vertical resistive force vertical displacement during cylinder indentation.

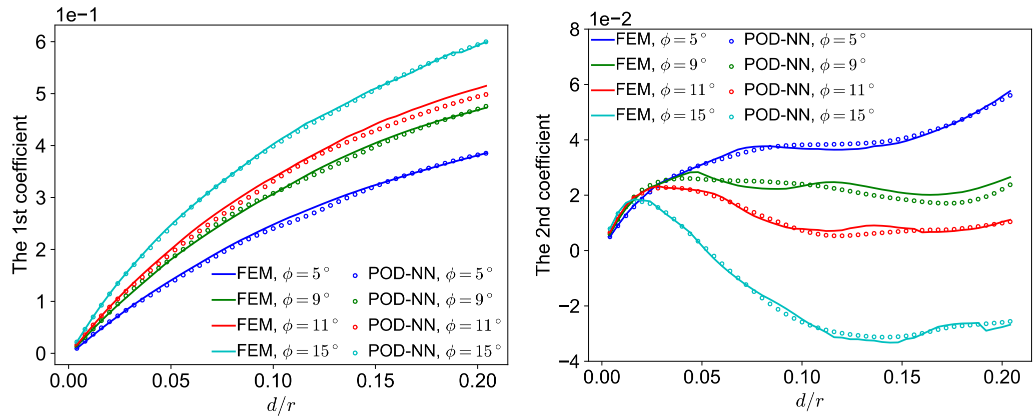
3.4. Plate ploughing

Ploughing processes of plates were examined in the last example. During a ploughing process, as shown in Fig. 1c, a plate was embedded in soil with an initial depth d^0 and then moved laterally or along an inclined direction α.

Finite element solutions for $\alpha = 0°, 5°, 10°, 20°, 30°, 40°, 50°, 60°$ were collected to train the neural network. In each RITSS-based FEM, the ratio of final horizontal displacement to the initial depth d^0 of the plate is around 1. 100 data points were collected from each simulation. 4 reduced basis vectors for surface geometries were generated (Fig. 15) after POD. Hence, the inputs of the neural network are the plates' inclination of motion α and 4 coefficients of reduced basis λ_1, λ_2, λ_3, λ_4.

3.4.1. Constant direction

After training the neural network, tests were conducted for 4 ploughing processes moving in different directions ($\alpha = 2.5°, 15°, 25°, 45°$). Figure 16 compares the horizontal forces as predicted by the neural network with those computed via FEM. Similar to the indentation results, the findings indicate that the data-driven model gives accurate predictions for force-displacement relations during ploughing. Figure 17 and 18 compare the 4 reduced basis coefficients and reconstructed free surfaces, where the errors of coefficients predicted by the neural network increase with the plates moving forward, leading to small discrepancies in the free surface geometries ($u/d^0 = 0.75$ and 1 in Fig. 18). The reason is that the coefficients serve as both the



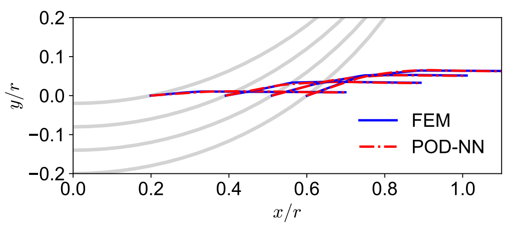
(a) (b)

Figure 13: Comparison of the coefficients of reduced basis during cylinder indentation.

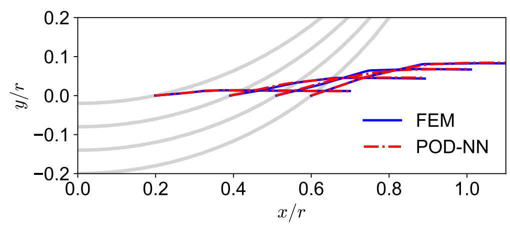
outputs at the current step as well as the inputs at the next step, causing errors to propagate. For $\alpha = 2.5°$ and $15°$, increases in the force occur when u/d^0 exceeds 0.8. This is consistent with the error propagation observed in the 1st coefficient. Despite the variation in error, the predicted forces remain close to FEM solutions throughout the whole process.

3.4.2. Arbitrary paths

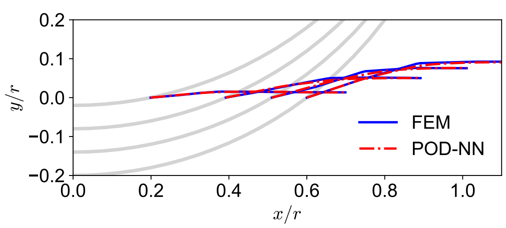
In previous test examples, the plate moved with a constant velocity along a fixed direction. For the following tests, four arbitrary motions were examined. Figure 19 illustrates the changes in the velocity direction during the ploughing for these four paths. Figure 20 compares predicted forces and numerically solved forces. Figure 21 and 22 compare the four reduced basis coefficients and surface geometries during ploughing, respectively. It is shown that the training model can still give good predictions for arbitrary paths, although the plates moved along a fixed direction in the training data. In terms of Path 1, the errors in the coefficients and reconstructed surface geometry increase with the plate moving forward. The same trend also occurs for errors in the force. In terms of Path 2, apparent errors appear when u/d^0 is greater than 0.7 for both the 1st coefficient and the force. Regarding Paths 3 and 4, there are apparent errors in the 1st coefficient, surface geometry, and force when $u/d^0 = 0.75$, and the errors become small when $u/d^0 = 1$. In summary, the results show that the errors in evolutionary free surface affect the accuracy of force-displacement relations.



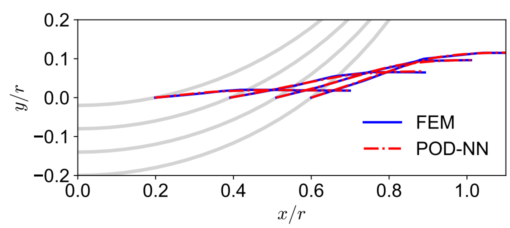
(a) $\phi = 5^\circ$



(b) $\phi = 9^\circ$

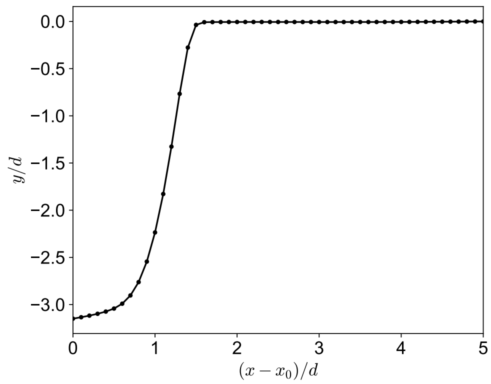


(c) $\phi = 11^\circ$

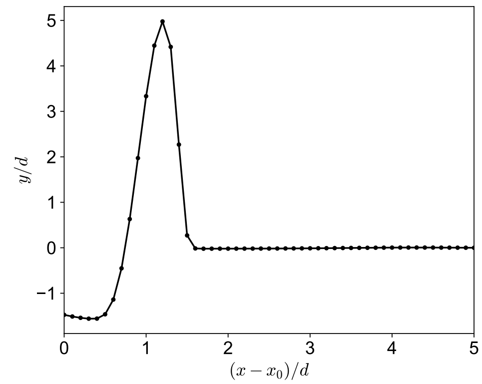


(d) $\phi = 15^\circ$

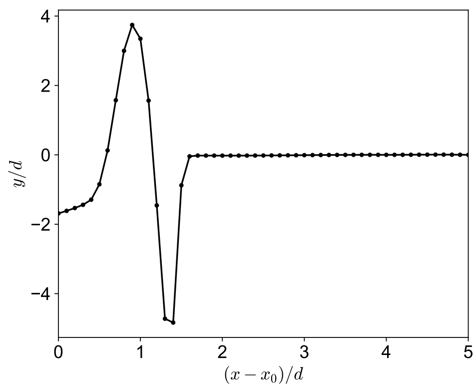
Figure 14: Comparison of free surfaces during cylinder indentation. The surface geometries are shown when the indentation depths are $0.02r$, $0.08r$, $0.14r$, and $0.20r$. The gray lines are the locations of cylinders, motion from top to bottom.



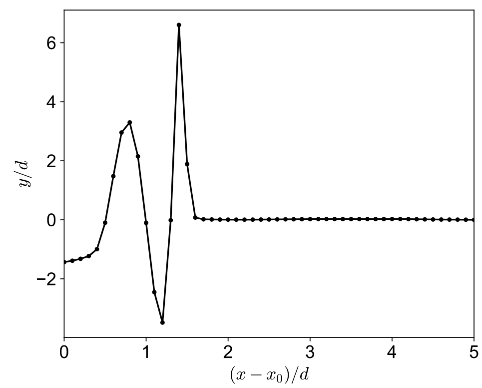
(a) The 1st reduced basis vector $\boldsymbol{v}_1$



(b) The 2nd reduced basis vector $\boldsymbol{v}_2$

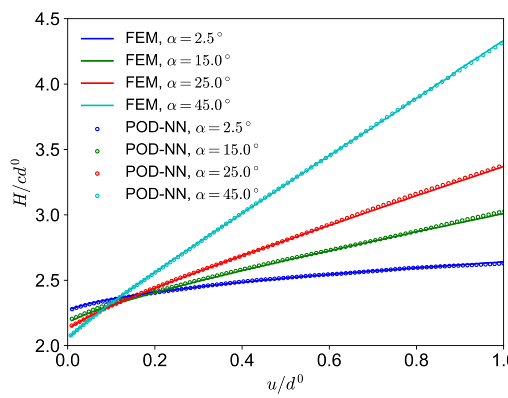


(c) The 3rd reduced basis vector $\boldsymbol{v}_3$

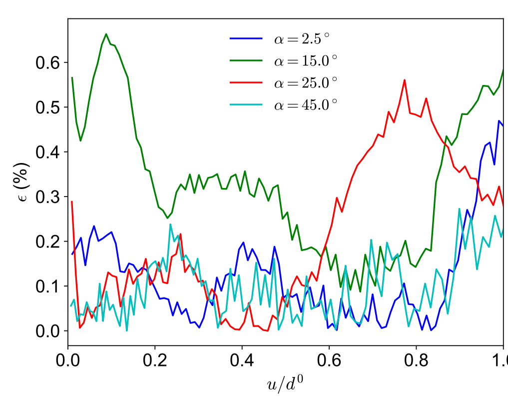


(d) The 4th reduced basis vector $\boldsymbol{v}_4$

Figure 15: Reduced basis for surface geometries during plate ploughing.



(a) Comparison of forces



(b) Relative errors of forces

Figure 16: Horizontal resistive force versus horizontal displacement for plates moving in constant directions.

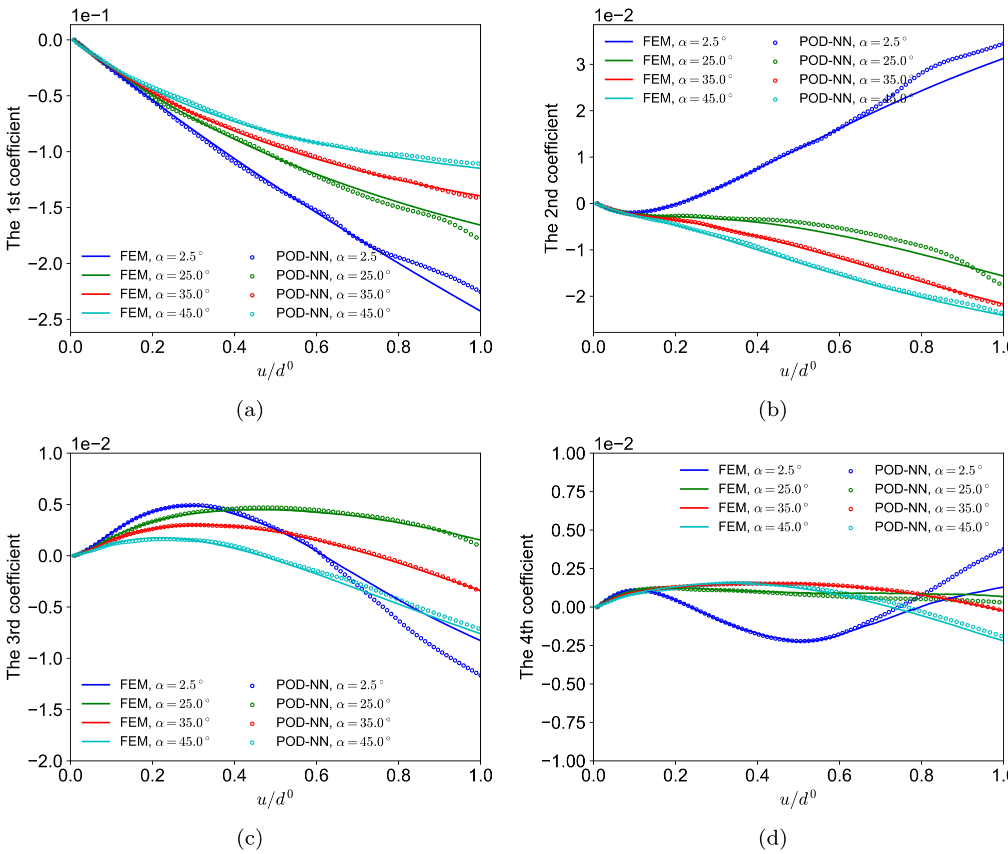
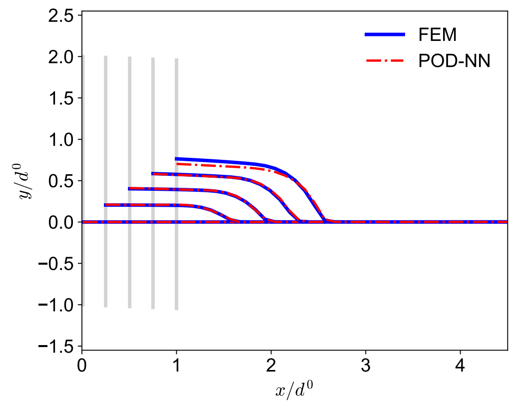
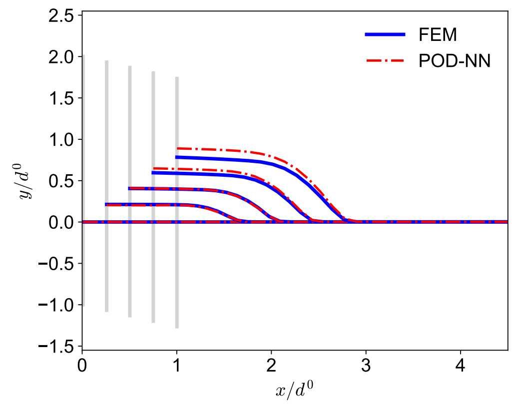


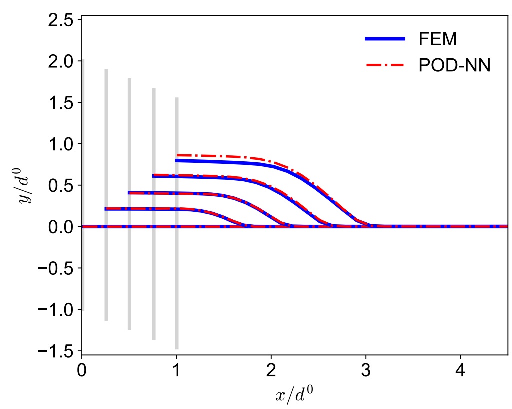
Figure 17: Comparison of the coefficients of reduced basis for plates moving in constant directions.



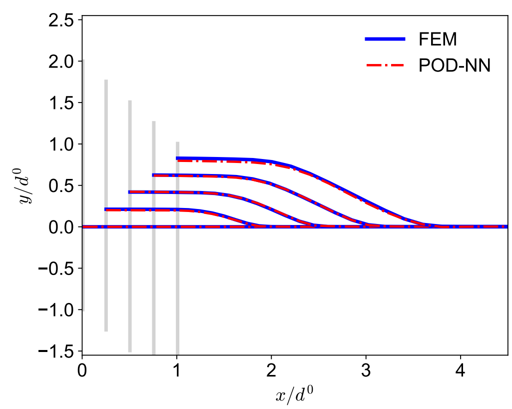
(a) $\alpha = 2.5°$



(b) $\alpha = 15°$



(c) $\alpha = 25°$



(d) $\alpha = 45°$

Figure 18: Comparison of the free surfaces for plates moving in constant directions. The surface geometries are shown when $u/d^0 = 0$, 0.25, 0.5, 0.75, and 1. The gray lines are the locations of plates, moving from left to right.

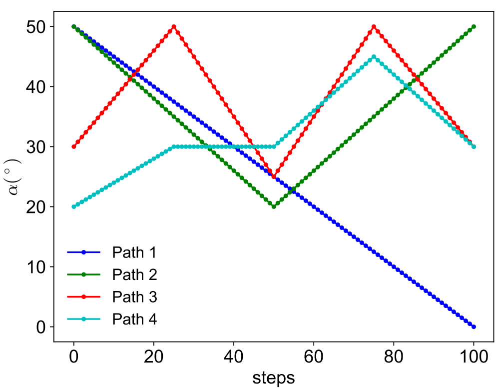
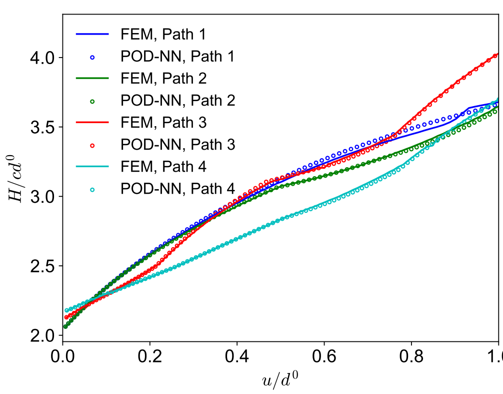
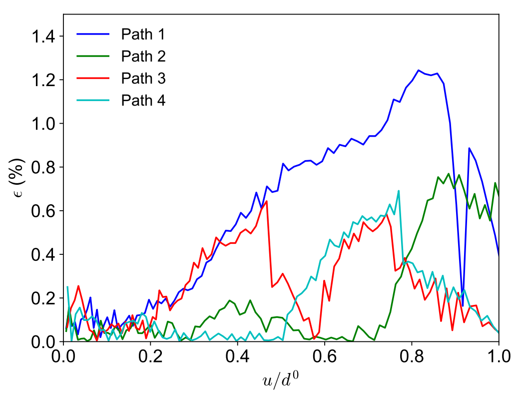


Figure 19: 4 paths with sequentially varied directions of velocity.



(a) Comparison of forces



(b) Relative errors of forces

Figure 20: Horizontal resistive force versus horizontal displacement for plates moving along arbitrary paths.

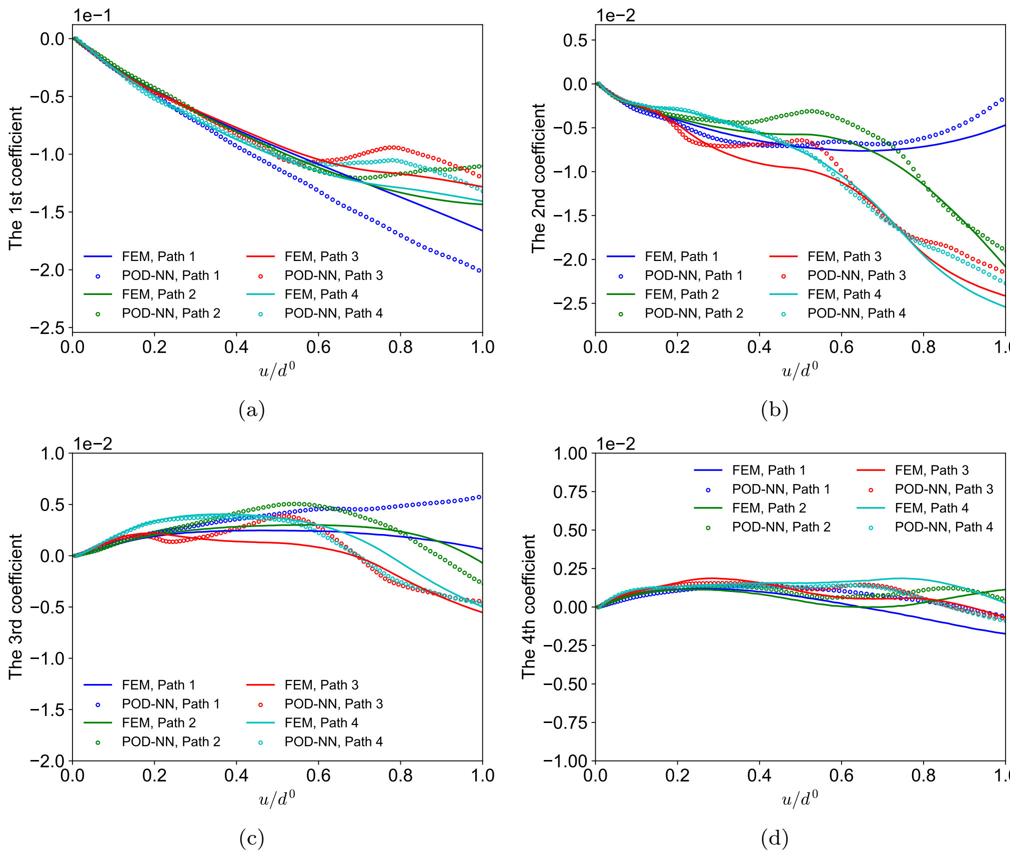
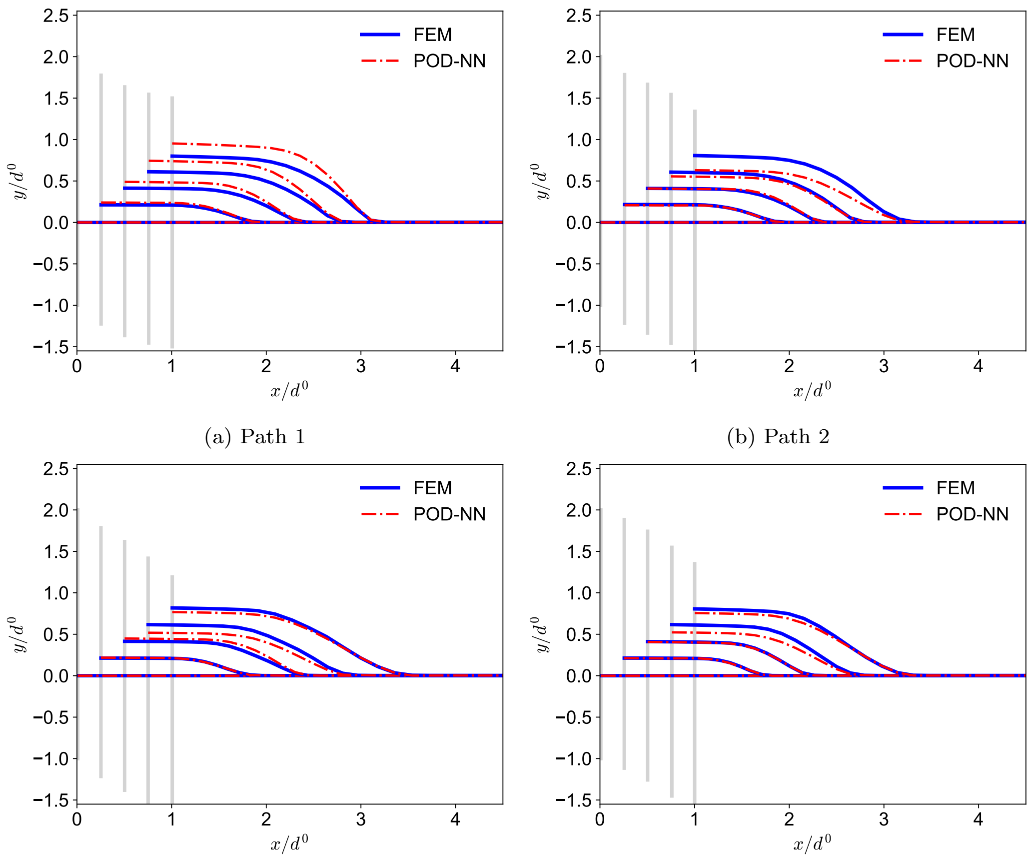


Figure 21: Comparison of the coefficients of reduced basis for plates moving along arbitrary paths.



(a) Path 1

(b) Path 2

(c) Path 3

(d) Path 4

Figure 22: Comparison of the free surfaces for plates moving along arbitrary paths. The surface geometries are shown when $u/d^0 = 0$, 0.25, 0.5, 0.75, and 1. The gray lines are the locations of plates, moving from left to right.

4. Discussion

The results show that the proposed data-driven approach effectively model evolving geometries and force-displacement histories. There is no significant error propagation as the objects advance. In the case of plate ploughing, although the trained data comes from plates moving in constant directions, the methods also exhibit strong performance in predicting the behavior of plates moving in changing directions. This highlights the generalizability of the approach.

Geometries are both inputs and outputs of neural networks in this proposed approach, which incorporate sequential steps to predict forces based on current geometries. The geometries can be regarded as memories of previous motions, similar to how plastic strains serve as a record of load histories in plastic constitutive models. This work primarily focuses on perfectly plastic materials, but future research could extend the proposed approach to encompass more complex constitutive laws. To effectively account for hardening/softening behaviors, for example, geometries might not be the only factors serving as memories of previous motions. It is valuable to investigate whether other factors, such as stress or strain fields around objects, should also be taken as inputs and outputs of the neural network, along with the surface geometries.

When it comes to approximating geometries, proper orthogonal decomposition (POD) demonstrates better performance compared to piecewise linear approximation, primarily because it utilizes data properties. Unlike piecewise linear methods, which may introduce artifacts due to their reliance on fixed segments, POD effectively captures the underlying patterns in the data, ensuring that the approximation results are consistently constrained within the bounds of the actual data during the whole process. In contrast, the errors associated with piecewise linear approximation tend to increase in this process.

5. Conclusion

This work explores the potential of data-driven methods in modeling soil-structure/tool interaction. The proposed method can predict the force-displacement relations during the whole process involving large deformation. Through the use of pre-computed data sets based on numerical simulations, the method is computationally efficient and practical for design optimization. To handle the large deformation and evolution of free surfaces, surface geometries are regarded as memories of previous motions and taken as both the inputs and outputs of the neural network. To handle the "curse of dimensionality", the degrees of freedom of surface geometries are reduced through proper orthogonal decomposition. The memorization of changing surface geometries and the dimensionality reduction of surfaces are key novelties of our proposed method. This efficient representation of changing geometries reduces the size of inputs and outputs of the neural network, thereby avoiding unmanageable computational time required to generate large amounts of data.

The capability of the proposed method is examined for three fundamental processes: wedge indentation, cylinder indentation, and plate ploughing. The comparison with the results from existing numerical methods shows that the data-driven method can predict the mechanical behaviors of soil-structure/tool interaction, accounting for various geometries of objects, soil parameters, and motions. The versatility of the method makes it applicable to a wide array of problems involving evolving geometries.

6. Acknowledgements

The authors are grateful for the support from the National Science Foundation under Grant No. 1846817. This work is derived from the first author's PhD thesis (Yang, 2024).

References

Abdi, H., Williams, L.J., 2010. Principal component analysis. Wiley interdisciplinary reviews: computational statistics 2, 433–459.

Benner, P., Gugercin, S., Willcox, K., 2015. A survey of projection-based model reduction methods for parametric dynamical systems. SIAM review 57, 483–531.

Bessa, M.A., Bostanabad, R., Liu, Z., Hu, A., Apley, D.W., Brinson, C., Chen, W., Liu, W.K., 2017. A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality. Computer Methods in Applied Mechanics and Engineering 320, 633–667.

Bienen, B., Byrne, B., Houlsby, G., Cassidy, M., 2006. Investigating six-degree-of-freedom loading of shallow foundations on sand. Géotechnique 56, 367–379.

Brownlee, J., 2019. A gentle introduction to the rectified linear unit (relu). Machine learning mastery 6.

Castells, F., Laguna, P., Sörnmo, L., Bollmann, A., Roig, J.M., 2007. Principal component analysis in ecg signal processing. EURASIP Journal on Advances in Signal Processing 2007, 1–21.

Chatterjee, S., White, D., Randolph, M., 2012. Numerical simulations of pipe–soil interaction during large lateral movements on clay. Géotechnique 62, 693–705.

Chatzigogos, C.T., Pecker, A., Salençon, J., 2009. Macroelement modeling of shallow foundations. Soil Dynamics and Earthquake Engineering 29, 765–781. URL: http://www.sciencedirect.com/science/article/pii/S0267726108001383, doi:10.1016/j.soildyn.2008.08.009.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems 2, 303–314.

Dijkstra, J., Broere, W., Heeres, O., 2011. Numerical simulation of pile installation. Computers and Geotechnics 38, 612–622.

Eckart, C., Young, G., 1936. The approximation of one matrix by another of lower rank. Psychometrika 1, 211–218.

Ghaboussi, J., Garrett Jr, J., Wu, X., 1991. Knowledge-based modeling of material behavior with neural networks. Journal of engineering mechanics 117, 132–153.

Ghaboussi, J., Pecknold, D.A., Zhang, M., Haj-Ali, R.M., 1998. Autoprogressive training of neural network constitutive models. International Journal for Numerical Methods in Engineering 42, 105–126.

Hamann, T., Qiu, G., Grabe, J., 2015. Application of a Coupled Eulerian–Lagrangian approach on pile installation problems under partially drained conditions. Computers and Geotechnics 63, 279–290. URL: https://www.sciencedirect.com/science/article/pii/S0266352X1400189X, doi:10.1016/J.COMPGEO.2014.10.006.

Hambleton, J., Drescher, A., 2008. Modeling wheel-induced rutting in soils: Indentation. Journal of Terramechanics 45, 201–211.

Hambleton, J., Drescher, A., 2009. Modeling wheel-induced rutting in soils: Rolling. Journal of Terramechanics 46, 35–47.

Hambleton, J., Drescher, A., 2012. Approximate model for blunt objects indenting cohesive-frictional materials. International Journal for Numerical and Analytical Methods in Geomechanics 36, 249–271.

Hambleton, J.P., Stanier, S.A., White, D.J., Sloan, S.W., 2014. Modelling ploughing and cutting processes in soils. Australian Geomechanics 49, 147–156.

Hashash, Y., Jung, S., Ghaboussi, J., 2004. Numerical implementation of a neural network based material model in finite element analysis. International Journal for numerical methods in engineering 59, 989–1005.

Hesthaven, J., Ubbiali, S., 2018. Non-intrusive reduced order modeling of nonlinear problems using neural networks. Journal of Computational Physics 363, 55–78. URL: https://www.sciencedirect.com/science/article/pii/S0021999118301190, doi:10.1016/J.JCP.2018.02.037.

Houlsby, G.T., Cassidy, M.J., 2002. A plasticity model for the behaviour of footings on sand under combined loading. Géotechnique 52, 117–129. URL: http://www.icevirtuallibrary.com/doi/10.1680/geot.2002.52.2.117, doi:10.1680/geot.2002.52.2.117.

Hu, Y., Randolph, M., 1998. A practical numerical approach for large deformation problems in soil. International Journal for Numerical and Analytical Methods in Geomechanics 22, 327–350.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Kong, D., Martin, C.M., Byrne, B.W., 2017. Modelling large plastic deformations of cohesive soils using sequential limit analysis. International Journal for Numerical and Analytical Methods in Geomechanics 41, 1781–1806. URL: http://doi.wiley.com/10.1002/nag.2700, doi:10.1002/nag.2700.

Le, B., Yvonnet, J., He, Q.C., 2015. Computational homogenization of nonlinear elastic materials using neural networks. International Journal for Numerical Methods in Engineering 104, 1061–1084.

Liu, Z., Wu, C., Koishi, M., 2019. A deep material network for multiscale topology learning and accelerated nonlinear modeling of heterogeneous materials. Computer Methods in Applied Mechanics and Engineering 345, 1138–1168.

McKyes, E., Ali, O., 1977. The cutting of soil by narrow blades. Journal of Terramechanics 14, 43–58.

Mozaffar, M., Bostanabad, R., Chen, W., Ehmann, K., Cao, J., Bessa, M., 2019. Deep learning predicts path-dependent plasticity. Proceedings of the National Academy of Sciences 116, 26414–26420.

Nova, R., Montrasio, L., 1991. Settlements of shallow foundations on sand. Géotechnique 41, 243–256. URL: http://www.icevirtuallibrary.com/doi/10.1680/geot.1991.41.2.243, doi:10.1680/geot.1991.41.2.243.

Qiu, G., Henke, S., Grabe, J., 2011. Application of a Coupled Eulerian–Lagrangian approach on geomechanical problems involving large deformations. Computers and Geotechnics 38, 30–39. URL: https://www.sciencedirect.com/science/article/pii/S0266352X10001175, doi:10.1016/J.COMPGEO.2010.09.002.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics 378, 686–707.

Rosenfeld, A., 1976. Digital picture processing. Academic press.

Rowley, C.W., Colonius, T., Murray, R.M., 2004. Model reduction for compressible flows using pod and galerkin projection. Physica D: Nonlinear Phenomena 189, 115–129.

Rozza, G., Huynh, D.B.P., Patera, A.T., 2008. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Archives of Computational Methods in Engineering 15, 229–275.

Shi, Z., Huang, M., Hambleton, J.P., 2021. Possibilities and limitations of the sequential kinematic method for simulating evolutionary plasticity problems. Computers and Geotechnics 140, 104449. URL: https://www.sciencedirect.com/science/article/pii/S0266352X21004377, doi:10.1016/j.compgeo.2021.104449.

Shmulevich, I., Asaf, Z., Rubinstein, D., 2007. Interaction between soil and a wide cutting blade using the discrete element method. Soil and Tillage Research 97, 37–50. URL: http://www.sciencedirect.com/science/article/pii/S0167198707001249, doi:10.1016/j.still.2007.08.009.

Sirovich, L., 1987. Turbulence and the dynamics of coherent structures. i. coherent structures. Quarterly of applied mathematics 45, 561–571.

Soga, K., Alonso, E., Yerro, A., Kumar, K., Bandara, S., 2015. Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method. Géotechnique 66, 248–273. URL: https://www.icevirtuallibrary.com/doi/full/10.1680/jgeot.15.LM.005, doi:10.1680/jgeot.15.LM.005. publisher: ICE Publishing.

Systèmes, D., 2016. Abaqus 2016 documentation. Dassault Systèmes .

Tian, Y., Cassidy, M.J., 2011. Pipe-soil interaction model incorporating large lateral displacements in calcareous sand. Journal of geotechnical and geoenvironmental engineering 137, 279–287.

Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) 58, 267–288.

Tsuji, T., Nakagawa, Y., Matsumoto, N., Kadono, Y., Takayama, T., Tanaka, T., 2012. 3-d dem simulation of cohesive soil-pushing behavior by bulldozer blade. Journal of Terramechanics 49, 37–47.

Wang, D., Bienen, B., Nazem, M., Tian, Y., Zheng, J., Pucker, T., Randolph, M.F., 2015. Large deformation finite element analyses in geotechnical engineering. Computers and Geotechnics 65, 104–114.

Wong, D., White, D.J., Randolph, M.F., 2010. Large-deformation finite element analysis of pipe penetration and large-amplitude lateral displacement. Canadian Geotechnical Journal 47, 842–856.

Yang, Q., 2024. Surrogate Modeling and Inverse Analysis for Soil Plasticity With Machine Learning. Ph.D. thesis. Northwestern University.

Yong, R., Hanna, A., 1977. Finite element analysis of plane soil cutting. Journal of terramechanics 14, 103–125.

Zhang, P., Yin, Z.Y., Zheng, Y., Gao, F.P., 2020. A lstm surrogate modelling approach for caisson foundations. Ocean Engineering 204, 107263.